

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

Applicant: Tatsunobu ANDO  
International Application No.: PCT/JP03/15647  
International Filing Date: December 8, 2003  
For: FINGERPRINT COLLATION APPARATUS AND  
METHOD, RECORDING MEDIUM AND PROGRAM


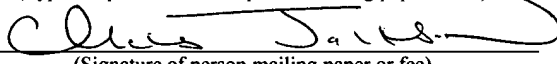
745 Fifth Avenue  
New York, NY 10151

**EXPRESS MAIL**

Mailing Label Number: EV723350595US

Date of Deposit: July 5, 2005

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" Service under 37 CFR 1.10 on the date indicated above and is addressed to Mail Stop PCT, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

  
(Typed or printed name of person mailing paper or fee)  
  
(Signature of person mailing paper or fee)

**CLAIM OF PRIORITY UNDER 37 C.F.R. § 1.78(a)(2)**


Mail Stop PCT  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

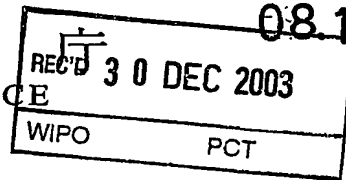
Sir:

Pursuant to 35 U.S.C. 119, this application is entitled to a claim of priority to Japanese Application No. 2003-000120 filed on January 6, 2003.

Respectfully submitted,

FROMMER LAWRENCE & HAUG LLP  
Attorneys for Applicant

By:   
William S. Frommer  
Reg. No. 25,506  
Tel. (212) 588-0800

日本国特許  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日  
Date of Application: 2003年 1月 6日

出願番号  
Application Number: 特願2003-000120  
[ST. 10/C]: [JP 2003-000120]

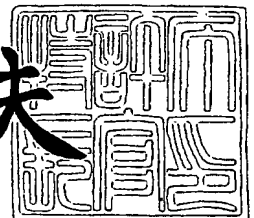
出願人  
Applicant(s): ソニー株式会社

**PRIORITY  
DOCUMENT**  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH RULE 17.1 (a) OR (b)

2003年10月28日

特許庁長官  
Commissioner,  
Japan Patent Office

今井康夫



【書類名】 特許願

【整理番号】 0290605308

【提出日】 平成15年 1月 6日

【あて先】 特許庁長官殿

【国際特許分類】 G06T 7/00

【発明者】

    【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社  
                                内

    【氏名】 安藤 辰伸

【特許出願人】

    【識別番号】 000002185

    【氏名又は名称】 ソニー株式会社

【代理人】

    【識別番号】 100082131

    【弁理士】

    【氏名又は名称】 稲本 義雄

    【電話番号】 03-3369-6479

【手数料の表示】

    【予納台帳番号】 032089

    【納付金額】 21,000円

【提出物件の目録】

    【物件名】 明細書 1

    【物件名】 図面 1

    【物件名】 要約書 1

    【包括委任状番号】 9708842

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 指紋照合装置および方法、記録媒体、並びに、プログラム

【特許請求の範囲】

【請求項 1】 指紋画像を処理する情報処理装置において、

前記指紋画像から指紋の分岐点と端点のうち一方である第 1 の特徴点を検出する第 1 の検出手段と、

複数の前記第 1 の特徴点のうち、近接する任意の 3 個を結ぶ 3 角形を作成する第 1 の作成手段と、

前記第 1 の作成手段により作成された前記 3 角形の面積と辺の長さを計算する第 1 の計算手段と、

前記第 1 の計算手段により計算された前記 3 角形の面積と辺の長さを記憶する記憶手段と

を備えることを特徴とする情報処理装置。

【請求項 2】 前記指紋画像の中心である中心点を検出する第 2 の検出手段と、

前記第 2 の検出手段により検出された前記中心点と複数の前記第 1 の特徴点の間の距離に基づいて、前記第 1 の特徴点をソートするソート手段と

をさらに備え、

前記第 1 の作成手段は、ソートされた前記第 1 の特徴点を利用して、近接する任意の 3 個を結ぶ 3 角形を作成する

ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】 前記第 1 の検出手段は、前記分岐点と前記端点のうちの他方である第 2 の特徴点をさらに検出し、

1 つの前記 3 角形を構成する 3 つの前記第 1 の特徴点をそれぞれ第 1 の点、第 2 の点、および第 3 の点とするとき、前記第 1 の点と前記第 1 の点に最も近い前記第 2 の特徴点である第 4 の点、前記第 2 の点と前記第 2 の点に最も近い前記第 2 の特徴点である第 5 の点、および、前記第 3 の点と前記第 3 の点に最も近い前記第 2 の特徴点である第 6 の点の、それぞれの距離と方向の少なくともいずれか一方を計算する第 2 の計算手段をさらに備え、

前記記憶手段は、前記第2の計算手段により計算された、前記第1の点と前記第4の点、前記第2の点と前記第5の点、および前記第3の点と前記第6の点のそれぞれの距離と方向の少なくともいずれか一方をさらに記憶する

ことを特徴とする請求項1に記載の情報処理装置。

【請求項4】 前記指紋画像の中心である中心点を検出する第2の検出手段と、

前記第2の検出手段により検出された前記中心点と複数の前記第1の特徴点の間の距離に基づいて、前記第2の特徴点をソートするソート手段と

をさらに備え、

前記第2の計算手段は、ソートされた前記第2の特徴点を利用して、前記第1の点乃至前記第3の点に対応する前記第4の点乃至前記第6の点の距離と方向の少なくともいずれか一方を計算する

ことを特徴とする請求項3に記載の情報処理装置。

【請求項5】 照合対象の指紋画像から指紋の前記第1の特徴点を検出する第2の検出手段と、

前記照合対象の指紋画像の指紋の複数の前記第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第2の作成手段と、

前記第2の作成手段により作成された前記3角形の面積と辺の長さを計算する第2の計算手段と、

前記記憶手段により記憶された、前記3角形の面積と辺の長さ、並びに、前記第2の計算手段により計算された、前記照合対象の指紋画像の前記3角形の面積と辺の長さを比較する比較手段と

をさらに備えることを特徴とする請求項1に記載の情報処理装置。

【請求項6】 指紋画像を処理する情報処理装置の情報処理方法において、前記指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、

複数の前記第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、

前記第1の作成ステップの処理により作成された前記3角形の面積と辺の長さ

を計算する第1の計算ステップと、

前記第1の計算ステップの処理により計算された前記3角形の面積と辺の長さの記憶を制御する記憶制御ステップと

を含むことを特徴とする情報処理方法。

【請求項7】 指紋画像を処理するプログラムであって、

前記指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、

複数の前記第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、

前記第1の作成ステップの処理により作成された前記3角形の面積と辺の長さを計算する第1の計算ステップと、

前記第1の計算ステップの処理により計算された前記3角形の面積と辺の長さの記憶を制御する記憶制御ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項8】 指紋画像を処理するプログラムであって、

前記指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、

複数の前記第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、

前記第1の作成ステップの処理により作成された前記3角形の面積と辺の長さを計算する第1の計算ステップと、

前記第1の計算ステップの処理により計算された前記3角形の面積と辺の長さの記憶を制御する記憶制御ステップと

を含む処理をコンピュータに実行させることを特徴とするプログラム。

【請求項9】 指紋画像を処理する情報処理装置において、

照合対象の前記指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出手段と、

複数の前記第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する

第1の作成手段と、

前記第1の作成手段により作成された前記3角形の面積と辺の長さを計算する第1の計算手段と、

前記第1の計算手段により計算された照合対象の前記指紋画像の前記3角形の面積と辺の長さを、予め記憶されている前記指紋画像の3角形の面積と辺の長さと比較する比較手段と

を備えることを特徴とする情報処理装置。

【請求項10】 前記第1の検出手段は、前記分岐点と前記端点のうちの他方である第2の特徴点をさらに検出し、1つの前記3角形を構成する3つの前記第1の特徴点をそれぞれ第1の点、第2の点、および第3の点とするとき、前記第1の点と前記第1の点に最も近い第2の特徴点である第4の点、前記第2の点と前記第2の点に最も近い第2の特徴点である第5の点、および、前記第3の点と前記第3の点に最も近い第2の特徴点である第6の点のそれぞれの距離と方向の少なくともいずれか一方を計算する第2の計算手段をさらに備え、

前記比較手段は、前記第1の計算手段と前記第2の計算手段により計算された照合対象の前記指紋画像の前記3角形の面積と辺の長さ、並びに前記第1の点に対する前記第4の点、前記第2の点に対する前記第5の点、および前記第3の点に対する前記第6の点のそれぞれの距離と方向の少なくともいずれか一方と、記憶されている前記指紋画像の3角形の面積と辺の長さ、並びに照合対象の指紋画像の前記第1の点に対する前記第4の点、前記第2の点に対する前記第5の点、および前記第3の点に対する前記第6の点のそれぞれの距離と方向の少なくともいずれか一方を比較する

ことを特徴とする請求項9に記載の情報処理装置。

【請求項11】 指紋画像を処理する情報処理装置の情報処理方法において

照合対象の前記指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、

複数の前記第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、

前記第 1 の作成ステップの処理により作成された前記 3 角形の面積と辺の長さを計算する第 1 の計算ステップと、

前記第 1 の計算ステップの処理により計算された照合対象の前記指紋画像の前記 3 角形の面積と辺の長さを、予め記憶されている前記指紋画像の 3 角形の面積と辺の長さと比較する比較ステップと

を含むことを特徴とする情報処理方法。

【請求項 12】 指紋画像を処理するプログラムであって、

照合対象の前記指紋画像から指紋の分岐点と端点のうち一方である第 1 の特徴点を検出する第 1 の検出ステップと、

複数の前記第 1 の特徴点のうち、近接する任意の 3 個を結ぶ 3 角形を作成する第 1 の作成ステップと、

前記第 1 の作成ステップの処理により作成された前記 3 角形の面積と辺の長さを計算する第 1 の計算ステップと、

前記第 1 の計算ステップの処理により計算された照合対象の前記指紋画像の前記 3 角形の面積と辺の長さを、予め記憶されている前記指紋画像の 3 角形の面積と辺の長さと比較する比較ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項 13】 指紋画像を処理するプログラムであって、

照合対象の前記指紋画像から指紋の分岐点と端点のうち一方である第 1 の特徴点を検出する第 1 の検出ステップと、

複数の前記第 1 の特徴点のうち、近接する任意の 3 個を結ぶ 3 角形を作成する第 1 の作成ステップと、

前記第 1 の作成ステップの処理により作成された前記 3 角形の面積と辺の長さを計算する第 1 の計算ステップと、

前記第 1 の計算ステップの処理により計算された照合対象の前記指紋画像の前記 3 角形の面積と辺の長さを、予め記憶されている前記指紋画像の 3 角形の面積と辺の長さと比較する比較ステップと

を含む処理をコンピュータに実行させることを特徴とするプログラム。



**【発明の詳細な説明】****【0001】****【発明の属する技術分野】**

本発明は、指紋照合装置および方法、記録媒体、並びに、プログラムに関し、特に、記憶するデータ量を少なくする指紋照合装置および方法、記録媒体、並びに、プログラムに関する。

**【0002】****【従来技術】**

使用者の指の指紋を読み取り、指紋を照合し、使用者であることを認証する指紋照合装置が各種の装置に応用されている。

**【0003】**

指の指紋を読み取り、指紋を登録する技術には、例えば、マニューシャ法やパターンマッチング法と呼ばれているものがある。

**【0004】**

マニューシャ法における読み取りの対象となる指紋の登録では、例えば、最初に、指紋画像を2値化して細線化が行なわれ、細線化像から端点や分岐点のような特徴点が抽出される。そして、特徴点から細線に沿って一定数の画素だけ遡り、この部分が部分細線として抽出され、この部分細線が近似する線分列に変換される。この操作が所定数の特徴点について繰り返されて、複数の線分の繋がりからなる線分列が抽出される。このようにして、指紋画像が線分列化され、その線分の両端の点の座標、線分が変わる位置の座標が登録される。また、特徴点が分岐点の時は、3本に枝分かれした細線について同様の作業が繰り返される。さらに、部分細線、細線化像、特徴点の種類および座標から、異なる部分細線の両端の点同士を結んだ線と交差する隆線の数が出算され、登録される（例えば、特許文献1参照）。

**【0005】**

また、マニューシャ法における指紋の照合では、最初に、指紋画像が2値化、細線化された後、特徴量が抽出される。そして、求めた細線化像の中で、登録された部分細線の1つと位置の近いものが選択され、両者のパターンが比較され、

不一致度がある閾値以下であれば、両者は同じものであるとされる。また、この位置合わせが照合画像中の特徴点について順次実行され、一致するものが検出されたら両者の位置合わせが行なわれ、このときの移動量だけ、他の部分細線が移動され、それぞれの部分細線について同様の比較が行なわれる。さらに、照合対象細線画像から交差隆線数が算出され、これと登録されている交差隆線数とが比較され、一致率が所定値以上であれば合格とみなされる（例えば、特許文献1参照）。

#### 【0006】

また、パターンマッチング法においては、指紋画像の全て、または、その一部が登録テンプレートとされる。

#### 【0007】

##### 【特許文献1】

特開平1-50175号公報

#### 【0008】

##### 【発明が解決しようとする課題】

しかしながら、パターンマッチング法は、指紋画像の全て、または、その一部を登録テンプレートとしているために、登録するテンプレートのサイズが大きなものとなり、その分、メモリの容量が必要となり、処理速度が遅くなるという課題があった。

#### 【0009】

また、取得した指紋画像が回転していた場合（例えば、上下が逆となっていた場合）、照合が容易ではなく、精度が十分でないという課題があった。

#### 【0010】

さらに、マニューシャ法（例えば、特許文献1参照）は、各特徴点の位置、方向、および種類（例えば、分岐点および端点）の情報に加えて、特徴点間の隆線数の情報を利用しており、特徴点間の隆線数の数により認証精度をあげている。すなわち、この方法においても、認証精度を上げるためには、ある程度の大きさのテンプレートサイズが必要になる。このように、処理速度がまだ遅い、データ量がまだ大きい、および、精度が充分ではないという課題があった。

## 【0011】

本発明は、このような状況に鑑みてなされたものであり、処理速度を速く、データ量を少なく、かつ、精度をあげることができるようにするものである。

## 【0012】

## 【課題を解決するための手段】

本発明の第1の情報処理装置は、指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出手段と、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成手段と、第1の作成手段により作成された3角形の面積と辺の長さを計算する第1の計算手段と、第1の計算手段により計算された3角形の面積と辺の長さを記憶する記憶手段とを備えることを特徴とする。

## 【0013】

前記指紋画像の中心である中心点を検出する第2の検出手段と、第2の検出手段により検出された中心点と複数の第1の特徴点の間の距離に基づいて、第1の特徴点をソートするソート手段とをさらに備え、第1の作成手段は、ソートされた第1の特徴点を利用して、近接する任意の3個を結ぶ3角形を作成するようにすることができる。

## 【0014】

前記第1の検出手段は、分岐点と端点のうちの他方である第2の特徴点をさらに検出し、1つの3角形を構成する3つの第1の特徴点をそれぞれ第1の点、第2の点、および第3の点とすると、第1の点と第1の点に最も近い第2の特徴点である第4の点、第2の点と第2の点に最も近い第2の特徴点である第5の点、および、第3の点と第3の点に最も近い第2の特徴点である第6の点の、それぞれの距離と方向の少なくともいずれか一方を計算する第2の計算手段をさらに備え、記憶手段は、第2の計算手段により計算された、第1の点と第4の点、第2の点と第5の点、および第3の点と第6の点のそれぞれの距離と方向の少なくともいずれか一方をさらに記憶するようにすることができる。

## 【0015】

前記指紋画像の中心である中心点を検出する第2の検出手段と、第2の検出手

段により検出された中心点と複数の第1の特徴点の間の距離に基づいて、第2の特徴点をソートするソート手段とをさらに備え、第2の計算手段は、ソートされた第2の特徴点を利用して、第1の点乃至第3の点に対応する第4の点乃至第6の点の距離と方向の少なくともいずれか一方を計算するようにすることができる。

#### 【0016】

前記照合対象の指紋画像から指紋の第1の特徴点を検出する第2の検出手段と、照合対象の指紋画像の指紋の複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第2の作成手段と、第2の作成手段により作成された3角形の面積と辺の長さを計算する第2の計算手段と、記憶手段により記憶された、3角形の面積と辺の長さ、並びに、第2の計算手段により計算された、照合対象の指紋画像の3角形の面積と辺の長さを比較する比較手段とをさらに備えるようにすることができる。

#### 【0017】

本発明の第1の情報処理方法は、指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、第1の作成ステップの処理により作成された3角形の面積と辺の長さを計算する第1の計算ステップと、第1の計算ステップの処理により計算された3角形の面積と辺の長さの記憶を制御する記憶制御ステップとを含むことを特徴とする。

#### 【0018】

本発明の第1の記録媒体に記録されているプログラムは、指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、第1の作成ステップの処理により作成された3角形の面積と辺の長さを計算する第1の計算ステップと、第1の計算ステップの処理により計算された3角形の面積と辺の長さの記憶を制御する記憶制御ステップとを含むことを特徴とする。

#### 【0019】

本発明の第1のプログラムは、指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、第1の作成ステップの処理により作成された3角形の面積と辺の長さを計算する第1の計算ステップと、第1の計算ステップの処理により計算された3角形の面積と辺の長さの記憶を制御する記憶制御ステップとを含む処理をコンピュータに実行させることを特徴とする。

#### 【0020】

本発明の第2の情報処理装置は、照合対象の指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出手段と、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成手段と、第1の作成手段により作成された3角形の面積と辺の長さを計算する第1の計算手段と、第1の計算手段により計算された照合対象の指紋画像の3角形の面積と辺の長さを、予め記憶されている指紋画像の3角形の面積と辺の長さと比較する比較手段とを備えることを特徴とする。

#### 【0021】

前記第1の検出手段は、分岐点と端点のうちの他方である第2の特徴点をさらに検出し、1つの3角形を構成する3つの第1の特徴点をそれぞれ第1の点、第2の点、および第3の点とすると、第1の点と第1の点に最も近い第2の特徴点である第4の点、第2の点と第2の点に最も近い第2の特徴点である第5の点、および、第3の点と第3の点に最も近い第2の特徴点である第6の点のそれぞれの距離と方向の少なくともいずれか一方を計算する第2の計算手段をさらに備え、比較手段は、第1の計算手段と第2の計算手段により計算された照合対象の指紋画像の3角形の面積と辺の長さ、並びに第1の点に対する第4の点、第2の点に対する第5の点、および第3の点に対する第6の点のそれぞれの距離と方向の少なくともいずれか一方と、記憶されている指紋画像の3角形の面積と辺の長さ、並びに照合対象の指紋画像の第1の点に対する第4の点、第2の点に対する第5の点、および第3の点に対する第6の点のそれぞれの距離と方向の少なくともいずれか一方を比較するようにすることができる。

**【0022】**

本発明の第2の情報処理方法は、照合対象の指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、第1の作成ステップの処理により作成された3角形の面積と辺の長さを計算する第1の計算ステップと、第1の計算ステップの処理により計算された照合対象の指紋画像の3角形の面積と辺の長さを、予め記憶されている指紋画像の3角形の面積と辺の長さと比較する比較ステップとを含むことを特徴とする。

**【0023】**

本発明の第2の記録媒体に記録されているプログラムは、照合対象の指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、第1の作成ステップの処理により作成された3角形の面積と辺の長さを計算する第1の計算ステップと、第1の計算ステップの処理により計算された照合対象の指紋画像の3角形の面積と辺の長さを、予め記憶されている指紋画像の3角形の面積と辺の長さと比較する比較ステップとを含むことを特徴とする。

**【0024】**

本発明の第2のプログラムは、照合対象の指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点を検出する第1の検出ステップと、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形を作成する第1の作成ステップと、第1の作成ステップの処理により作成された3角形の面積と辺の長さを計算する第1の計算ステップと、第1の計算ステップの処理により計算された照合対象の指紋画像の3角形の面積と辺の長さを、予め記憶されている指紋画像の3角形の面積と辺の長さと比較する比較ステップとを含む処理をコンピュータに実行させることを特徴とする。

**【0025】**

第1の本願発明においては、指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点が検出され、複数の第1の特徴点のうち、近接する任意の3個を

結ぶ3角形が作成され、作成された3角形の面積と辺の長さが計算され、計算された3角形の面積と辺の長さが記憶される。

#### 【0026】

第2の本願発明においては、照合対象の指紋画像から指紋の分岐点と端点のうち一方である第1の特徴点が検出され、複数の第1の特徴点のうち、近接する任意の3個を結ぶ3角形が作成され、作成された3角形の面積と辺の長さが計算され、計算された照合対象の指紋画像の3角形の面積と辺の長さが、予め記憶されている指紋画像の3角形の面積と辺の長さと比較される。

#### 【0027】

##### 【発明の実施の形態】

図1は、本発明を適用した指紋照合装置の一実施の形態の外観の構成を示す図である。

#### 【0028】

指紋照合装置1は、装置の筐体をなす本体部11、および閉じた状態において本体部11の上面を物理的に保護する、本体部11に対して開閉自在に設けられている保護カバー12からなる。図1においては、指紋照合装置1の保護カバー12は、本体部11に対して開かれた状態とされている。

#### 【0029】

本体部11の上面には、指紋読み取りセンサ21、並びに指紋の照合の結果を表示するランプ22およびランプ23が設けられている。

#### 【0030】

指紋読み取りセンサ21は、指紋読み取りセンサ21に押圧され、接触された指の指紋を画像として読み取る。

#### 【0031】

図1に示す状態において、図2に示すように、指紋読み取りセンサ21を、指紋が読み取られる指41で軽く押圧することにより、指41の指紋が画像として読み取られる。

#### 【0032】

ランプ22およびランプ23は、排他的に点灯し、即ち、これにより、指紋照

合の可否を表示する。ランプ 22 およびランプ 23 は、いずれも消灯することにより、指紋読み取りセンサ 21 に指 41 が接触されていない状態であることを表示する。

#### 【0033】

指紋照合装置 1 は、図 2 に示すように、指紋の照合の結果を示すデータを、端子 42 に接続されているケーブル 43 を介して、図示せぬ他の機器に供給する。ケーブル 43 は、例えば、USB (Universal Serial Bus) の規格に基づいたケーブルとすることができる。ケーブル 43 の規格は、例えば、RS232C など、他の規格であってもよいことは勿論である。

#### 【0034】

この指紋照合装置 1 を持ち運びするとき、保護カバー 12 は閉じられる。これにより、本体部 11、表示用のランプ 22、およびランプ 23、特に、指紋読み取りセンサ 21 が、外部からの異常な圧力から保護されるとともに、静電気から保護される。その結果、指紋照合装置 1 の破損または破壊が抑制される。

#### 【0035】

図 3 は、指紋照合装置 1 の内部の構成を示すブロック図である。

#### 【0036】

指紋読み取り用 LSI (Large Scale Integrated circuit) 51 は、指紋読み取りセンサ 21 から供給された信号から、指紋の画像を生成して、生成した指紋の画像を CPU 52 に供給する。

#### 【0037】

CPU 52 は、指紋照合装置 1 全体の動作を制御する。また、CPU 52 は、指紋読み取り用 LSI 51 から供給された指紋の画像を登録したり、既に登録されている指紋と照合する。

#### 【0038】

プログラム用 RAM (Random Access Memory) /ROM (Read Only Memory) 53 は、RAM および ROM からなり、CPU 52 が実行するコントロールファームウェアである、指紋照合プログラムを記憶する。プログラム用 RAM/ROM 53 の ROM 部分には、基本的に固定のプログラムおよびパラメータが記憶される。プログラム用 RAM/RO



M53のRAM部分には、指紋照合プログラムの実行に伴って変化するパラメータまたはデータが記憶される。

#### 【0039】

フラッシュメモリ54は、使用者の指紋を登録するとき、使用者の指紋の画像から抽出された特徴量を表わすデータを登録テンプレートとして記憶する。すなわち、CPU52は、指紋を登録するとき、指紋読み取り用LSI51から供給された、指紋の画像から抽出された登録テンプレートをフラッシュメモリ54に記憶させる。また、CPU52は、指紋を照合するとき、フラッシュメモリ54に記憶されている登録テンプレートと、指紋読み取り用LSI51から供給された指紋の画像とを照合する。

#### 【0040】

USBコントローラ55は、USBの規格に基づいて、端子42を介して接続されている、図示せぬ外部の機器に、CPU52から供給された、指紋照合の結果を出力すると共に、外部の機器から供給された、指紋を読み取っている使用者のID (Identifier) を取得する。USBコントローラ55は、取得した使用者のIDをCPU52に供給する。ユーザIDは、フラッシュメモリ54に、登録テンプレートとして記憶される。

#### 【0041】

指紋読み取り用LSI51、フラッシュメモリ54、プログラム用RAM/ROM53、CPU52、およびUSBコントローラ55は、バス56により相互に接続されている。

#### 【0042】

バス56には、必要に応じてドライブ57が接続される。ドライブ57は、必要に応じて装着される磁気ディスク71、光ディスク72、光磁気ディスク73、または半導体メモリ74に記録されているデータまたはプログラムを読み出して、そのデータまたはプログラムを、バス56を介してプログラム用RAM/ROM53に供給する。

#### 【0043】

次に、指紋読み取りセンサ21について説明する。指紋読み取りセンサ21は

、上述したように、指 4 1 をセンサ面に直接触れさせることで指紋画像を読み取るものであり、例えば、ソニー社製 C X A 3 2 7 1 G E を用いることができる。

#### 【0044】

指紋読み取りセンサ 2 1 においては、金属電極をアレイ状に並べた上面に絶縁膜を設けてセンサ面が構成される。指紋読み取りセンサ 2 1 のセンサ面に指を置いたとき、金属電極、絶縁膜、および指 4 1 の 3 つでコンデンサが形成される。この場合、指 4 1 は、電極として機能する。

#### 【0045】

指 4 1 の指紋の隆起部（電極）から指紋読み取りセンサ 2 1 の金属電極までの距離は、指 4 1 の指紋の谷部（電極）から指紋読み取りセンサ 2 1 の金属電極までの距離より短い。また、指 4 1 の指紋の隆起部が絶縁膜に接するので、指 4 1 の指紋の隆起部で形成されるコンデンサの容量値は、ほぼ、絶縁膜の誘電率で決まる。

#### 【0046】

これに対して、指 4 1 の指紋の谷部においては、電極である指 4 1 と絶縁膜との間に空気層が入るので、指 4 1 の指紋の谷部で形成されるコンデンサの容量値は、指 4 1 の指紋の隆起部で形成されるコンデンサの容量値と大きく異なる。

#### 【0047】

そこで、指紋読み取りセンサ 2 1 は、金属電極、絶縁膜、および指 4 1 の指紋の隆起部、並びに、金属電極、絶縁膜、および指 4 1 の指紋の谷部でそれぞれ形成されるコンデンサに蓄積される電荷を電圧に変換することで、指紋の凹凸を電気信号として出力する。

#### 【0048】

図 4 は、指紋読み取りセンサ 2 1 の原理的な構成を示す図である。

#### 【0049】

図 4 に示すように、指紋読み取りセンサ 2 1 は、シリコン (S i) 基板 1 0 1 上に層間膜 1 0 2 を介して金属電極 1 0 3 (サンプリング点) を  $80\mu\text{m}$  のピッチでアレイ状に配置し、その上面を絶縁膜 (Over Coat) 1 0 4 で覆った構成を有する。指紋読み取りセンサ 2 1 は、絶縁膜 1 0 4 の上面に直接置かれた指 4 1 の

指紋の凹凸を検出する。

#### 【0050】

すなわち、指41は導体なので、指41が絶縁膜104の上面に置かれたとき、金属電極103、絶縁膜104、および指41でコンデンサ106が形成される。そして、指紋の隆起部41Aおよび谷部41Bの凹凸により、電極である指41から金属電極103までの距離の異なるコンデンサ106が形成されることになり、指紋の隆起部41Aおよび谷部41Bの凹凸は、コンデンサ106の容量値の違いとなって現れる。

#### 【0051】

また、指紋の隆起部41Aは、絶縁膜104に接触しているので、隆起部41Aにより形成されるコンデンサ106の容量値は、絶縁膜104の誘電率および絶縁膜104の厚さで決まる。これに対して、指紋の谷部41Bにおいては、電極である指41と金属電極103との間に、絶縁膜104に加えて空気層が入るので、谷部41Bにより形成されるコンデンサ106の容量値は、絶縁膜104の誘電率および絶縁膜104の厚さ、並びに空気の誘電率およびその厚さで決まる。

#### 【0052】

すなわち、指41の指紋の谷部41Bで形成されるコンデンサ106の容量値と、指41の指紋の隆起部41Aで形成されるコンデンサ106の容量値とは、大きく異なる。

#### 【0053】

従って、指41が指紋読み取りセンサ21の上面（センサ面）に接触しているとき、金属電極103に一定電圧を印加すれば、指41の指紋の隆起部41Aのコンデンサ106に蓄積される電荷の量と、谷部41Bのコンデンサ106に蓄積される電荷の量は異なる。指紋読み取りセンサ21は、コンデンサ106に蓄積された電荷を電圧に変換して、指紋の凹凸を電気信号として出力する。

#### 【0054】

次に、指紋読み取りセンサ21の、指紋の凹凸を電気信号に変換する動作について説明する。図5は、指紋読み取りセンサ21の内部回路を示す回路図である

## 【0055】

指紋読み取りセンサ 21 は、指紋の凹凸による容量値を電荷として検出するセンサ部 21A、センサ部 21A から出力された電荷を電圧信号に変換するセンスアンプ部 21B、およびセンスアンプ部 21B から出力された電圧信号を増幅し、出力する出力部 21C から構成される。

## 【0056】

指紋読み取りセンサ 21 は、センスアンプ部 21B に、電荷を電圧信号に変換するための差動アンプ 121 を有し、出力部 21C に、電圧信号を増幅するための差動アンプ 122、および出力インピーダンスを調整するための差動アンプ 123 を有する。

## 【0057】

なお、図 5 において、コンデンサ  $C_s$  は、指 41 と金属電極 103 との間で形成されるコンデンサ 106 を示す。以下、コンデンサ  $C_s$  の容量は、単に、 $C_s$  と記述する。

## 【0058】

センサ部 21A のコンデンサ  $C_p$  は、金属電極 103 とシリコン基板 101 との間で形成される寄生容量に等価なコンデンサである。以下、コンデンサ  $C_p$  の容量は、単に、 $C_p$  と記述する。センサ部 21A のコンデンサ  $C_s$  およびコンデンサ  $C_p$  は、直列に接続される。コンデンサ  $C_s$  およびコンデンサ  $C_p$  の接続点の電圧を、 $V_{cel}$  と記述する。

## 【0059】

また、センスアンプ部 21B のコンデンサ  $C_{p'}$  は、寄生容量によるコンデンサ  $C_p$  の容量をキャンセルする目的のコンデンサである。コンデンサ  $C_{p'}$  の一方の端子は接地され、コンデンサ  $C_{p'}$  の他方の端子は、スイッチ  $Sw_3$  を介して、差動アンプ 121 の反転入力端子に接続されている。以下、コンデンサ  $C_{p'}$  の容量は、単に、 $C_{p'}$  と記述する。コンデンサ  $C_{p'}$  の容量は、コンデンサ  $C_p$  の容量にほぼ等しい。

## 【0060】

センスアンプ部 21B のコンデンサ Ch1 の一方の端子は接地され、コンデンサ Ch1 の他方の端子は、スイッチ Sw5 を介して差動アンプ 121 の出力端子に接続されると共に、スイッチ Sw6 を介して差動アンプ 122 の反転入力端子に接続されている。出力部 21C のコンデンサ Ch2 の一方の端子は接地され、コンデンサ Ch2 の他方の端子は、スイッチ Sw7 を介して、差動アンプ 122 の出力端子に接続されると共に、差動アンプ 123 の非反転入力端子に接続されている。センスアンプ部 21B のコンデンサ Ch1 および出力部 21C のコンデンサ Ch2 は、電圧を保持するための、いわゆるホールド容量である。

**【0061】**

以下、コンデンサ Ch1 の容量は、単に、Ch1 と記述する。コンデンサ Ch2 の容量は、単に、Ch2 と記述する。

**【0062】**

また、センスアンプ部 21B のコンデンサ Cf1 の一方の端子は、差動アンプ 121 の反転入力端子に接続され、コンデンサ Cf1 の他方の端子は、差動アンプ 121 の出力端子に接続されている。コンデンサ Cf1 の容量は、差動アンプ 121 のゲインを決める帰還容量である。

**【0063】**

以下、コンデンサ Cf1 の容量は、単に、Cf1 と記述する。

**【0064】**

出力部 21C のコンデンサ Cf2 の一方の端子は、差動アンプ 122 の反転入力端子に接続され、コンデンサ Cf2 の他方の端子は、差動アンプ 122 の出力端子に接続されている。コンデンサ Cf2 の容量は、差動アンプ 122 のゲインを決める帰還容量である。以下、コンデンサ Cf2 の容量は、単に、Cf2 と記述する。

**【0065】**

センスアンプ部 21B のスイッチ Swr は、コンデンサ Cs およびコンデンサ Cp の接続点と、差動アンプ 121 の反転入力端子とを接続するか、または切断する。センスアンプ部 21B のスイッチ Sw6 は、コンデンサ Ch1 の接地されていない端子と、差動アンプ 122 の反転入力端子とを接続するか、または切断

する。

#### 【0066】

センスアンプ部 21B において、スイッチ Sw1 の一方の端子には、電圧 VH が印加され、他方の端子は、差動アンプ 121 の非反転入力端子に接続される。スイッチ Sw2 の一方の端子には、電圧 VM が印加され、他方の端子は、差動アンプ 121 の非反転入力端子に接続される。スイッチ Sw11 の一方の端子には、電圧 VL が印加され、他方の端子は、差動アンプ 121 の非反転入力端子に接続される。

#### 【0067】

電圧 VH は、電圧 VM より高く、電圧 VM は、電圧 VL より高い。また電圧 VH と電圧 VM との差は、電圧 VM と電圧 VL との差にほぼ等しい。

#### 【0068】

スイッチ Sw3 の一方の端子は、コンデンサ Cp' の接地されていない端子に接続され、スイッチ Sw3 の他方の端子は、差動アンプ 121 の反転入力端子に接続されている。

#### 【0069】

スイッチ Sw4 の一方の端子は、コンデンサ Cf1 の一方の端子に接続され、スイッチ Sw4 の他方の端子は、コンデンサ Cf1 の他方の端子に接続されている。すなわち、スイッチ Sw4 は、コンデンサ Cf1 に並列に接続されている。

#### 【0070】

スイッチ Sw5 の一方の端子は、差動アンプ 121 の出力端子に接続され、スイッチ Sw5 の他方の端子は、コンデンサ Ch1 の接地されていない端子に接続されるとともに、スイッチ Swe を介して差動アンプ 122 の反転入力端子に接続されている。

#### 【0071】

出力部 21C のスイッチ Sw6 の一方の端子は、コンデンサ Cf2 の一方の端子に接続され、スイッチ Sw6 の他方の端子は、コンデンサ Cf2 の他方の端子に接続されている。すなわち、スイッチ Sw6 は、コンデンサ Cf2 に並列に接続されている。

## 【0072】

出力部 21C のスイッチ Sw7 の一方の端子は、差動アンプ 122 の出力端子に接続され、スイッチ Sw7 の他方の端子は、コンデンサ Ch2 の接地されていない端子に接続されると共に、差動アンプ 123 の非反転入力端子に接続されている。

## 【0073】

差動アンプ 122 の非反転入力端子には、電圧 VOS が印加されている。

## 【0074】

次に、指紋読み取りセンサ 21 の内部回路の動作について説明する。

## 【0075】

なお、スイッチ Swr、スイッチ Swe、スイッチ Sw1 乃至スイッチ Sw7、並びにスイッチ Sw11 が、切断している状態、すなわちオフしている状態から、指紋読み取りセンサ 21 の動作が開始される。

## 【0076】

(1) 初期状態、すなわち、スイッチ Swr、スイッチ Swe、スイッチ Sw1 乃至スイッチ Sw7、並びにスイッチ Sw11 が、オフしている状態から、まず、スイッチ Sw1、スイッチ Sw4、およびスイッチ Swr が、接続、すなわちオンされる。従って、コンデンサ Cs とコンデンサ Cp との接続点の電圧 Vce1 は、VH とされる。このとき、コンデンサ Cs およびコンデンサ Cp に蓄積される電荷は、 $(Cs + Cp) VH$  となる。

(2) その後、スイッチ Sw1、およびスイッチ Swr がオフされる。

(3) 次に、スイッチ Sw11、およびスイッチ Sw3 がオンされ、コンデンサ Cp' とスイッチ Sw3 の接続点の電圧 Vdmy を VL とする。ここで、コンデンサ Cp' に蓄積される電荷は、 $Cp' VL$  となる。

(4) その後、スイッチ Sw3、およびスイッチ Sw11 がオフされる。

(5) 次に、スイッチ Sw2 がオンされ、差動アンプ 121 の反転入力端子に印加される電圧 Vsl を VM とする。

(6) その後、スイッチ Sw4 がオフされる。

(7) 次に、スイッチ Swr、スイッチ Sw3、およびスイッチ Sw5 がオンさ

れる。このとき、電圧 $V_{ce1}$ は $V_H$ であり、電圧 $V_{dmy}$ は $V_L$ であり、差動アンプ121の反転入力端子に印加される電圧 $V_{s1}$ は $V_M$ なので、コンデンサ $C_s$ 、コンデンサ $C_p$ 、およびコンデンサ $C_{p'}$ 間で、式(1)で示される電荷が移動する。

【0077】

$$(C_s + C_p)(V_H - V_M) - C_{p'}(V_M - V_L) \\ \doteq C_s(V_H - V_M) \quad (1)$$

【0078】

従って、差動アンプ121の出力電圧 $V_{sns}$ は、式(2)で示される。

【0079】

$$V_{sns} = V_M - C_s(V_H - V_M) / C_{f1} \quad (2)$$

【0080】

このように、センスアンプ部21Bのゲインは、金属電極103とシリコン基板101との間で形成される寄生容量 $C_p$ に依存することなく決まるので、ダイナミックレンジを大きくすることができる。

【0081】

そして、コンデンサ $C_{h1}$ には、式(2)で求められる電圧 $V_{sns}$ が印加される。

【0082】

(8) その後、スイッチ $S_{w5}$ がオフされる。

(9) 次に、スイッチ $S_{w6}$ がオンされ、コンデンサ $C_{f2}$ の入力側(差動アンプ122の反転入力端子側)の電圧 $V_{oi}$ が $V_{OS}$ とされる。

(10) その後、スイッチ $S_{w6}$ がオフされる。

(11) 次に、スイッチ $S_{we}$ およびスイッチ $S_{w7}$ がオンされる。このとき、コンデンサ $C_{h1}$ からコンデンサ $C_{f2}$ に移動する電荷は、 $(V_{OS} - V_{sns})C_{h1}$ である。これにより、コンデンサ $C_{f2}$ の出力側(差動アンプ122の出力端子側)の電圧 $V_{oo}$ が決定し、コンデンサ $C_{h2}$ に電荷が蓄積される。

【0083】

そして、この電圧 $V_{oo}$ がバッファアンプである差動アンプ123を介して、



出力端子 A o u t に出力される。

【0084】

次に、図6と図7のフローチャートを参照して、図2の指紋照合装置1における指紋登録処理を説明する。なお、この処理は、ユーザにより、指紋登録処理を開始する指令が行なわれ、かつ、指紋読み取りセンサ21に、ユーザの指が押圧されたとき開始される。

【0085】

ステップS1において、指紋読み取りセンサ21は、指紋読み取りセンサ21に押圧され、接触された指の指紋の画像を読み取る。そして、読み取った指紋に対応する信号が、指紋読み取り用LSI51に供給される。

【0086】

ステップS2において、指紋読み取り用LSI51は、指紋読み取りセンサ21から供給された指紋に対応する信号から、指紋の画像データを生成する。

【0087】

ステップS3において、指紋読み取り用LSI51は、生成した指紋の画像データを、256階調のグレー画像データに変換し、変換した256階調のグレー画像データをCPU52に供給する。具体的には、図8に示されるような、128画素×128画素のグレー画像200のデータが取得される。画像201は、画像200の中の96画素×96画素の画像である。

【0088】

ステップS4において、CPU52は、変換されたグレー画像データに対して、ノイズ除去や指紋隆線内の微少な傷を埋めるなどの前処理を実行する。前処理が行なわれた画像は、図9に示されるように、ノイズが除去された画像となる。

【0089】

ステップS5において、CPU52は、前処理されたグレー画像データを、所定の閾値で2値化し、2値化画像データに変換する。図10は、2値化画像を表わしている。

【0090】

ステップS6において、CPU52は、細線化処理を行なう。具体的には、ステ

ステップS5の処理により得られた2値化画像のデータのうち、値が1である部分から線幅が1画素分の幅になるような中心線を抽出する処理が行なわれる。これにより、線幅が1画素分の幅となり、かつ、連結性が保持されるため、この後の指紋隆線の分岐点や端点を検出するのが容易になる。ステップS6の処理により細線化処理が行なわれた画像は、図11に示されるようになる。

#### 【0091】

ステップS7において、CPU52は、細線化処理された画像から、分岐点と端点をそれぞれ最大10個まで検出する。ステップS7の処理により検出された分岐点と端点は、図12に示されるようになる。検出される分岐点と端点は、実際には、図12の128画素×128画素の画像200のうち、96画素×96画素で示される画像201が対象となる。これにより抽出する分岐点と端点を減らすことができる。図12において、分岐点は、P1乃至P8（8個）とされ、分岐点P1乃至P8における座標は、それぞれ、P1（ $x_{p1}$ ,  $y_{p1}$ ）、P2（ $x_{p2}$ ,  $y_{p2}$ ）、P3（ $x_{p3}$ ,  $y_{p3}$ ）、P4（ $x_{p4}$ ,  $y_{p4}$ ）、P5（ $x_{p5}$ ,  $y_{p5}$ ）、P6（ $x_{p6}$ ,  $y_{p6}$ ）、P7（ $x_{p7}$ ,  $y_{p7}$ ）、およびP8（ $x_{p8}$ ,  $y_{p8}$ ）とされる。また、端点は、Q1乃至Q10（10個）とされ、端点Q1乃至Q10における座標は、それぞれ、Q1（ $x_{q1}$ ,  $y_{q1}$ ）、Q2（ $x_{q2}$ ,  $y_{q2}$ ）、Q3（ $x_{q3}$ ,  $y_{q3}$ ）、Q4（ $x_{q4}$ ,  $y_{q4}$ ）、Q5（ $x_{q5}$ ,  $y_{q5}$ ）、Q6（ $x_{q6}$ ,  $y_{q6}$ ）、Q7（ $x_{q7}$ ,  $y_{q7}$ ）、Q8（ $x_{q8}$ ,  $y_{q8}$ ）、Q9（ $x_{q9}$ ,  $y_{q9}$ ）およびQ10（ $x_{q10}$ ,  $y_{q10}$ ）とされる。

#### 【0092】

なお、いまの例の場合、CPU52は、検出する分岐点と端点を最大10個としたが、これに限らず、例えば、前もって決められた最大数のリミットとしてもよいし、10個以上と設定してもよい。また、CPU52は、検出する分岐点と端点の対象を、96画素×96画素で示される画像201としたが、これに限らず、例えば、112画素×112画素で構成される画像とすることもできる。

#### 【0093】

ステップS8において、CPU52は、プログラム用RAM/ROM53のRAM部分に、

検出した分岐点と端点の座標  $(x, y)$  (すなわち、 $P_i(x_{pi}, y_{pi})$  (ただし、 $i = 1, 2, \dots, 8$ )、および  $Q_i(x_{qi}, y_{qi})$  (ただし、 $i = 1, 2, \dots, 10$ )) を記憶させる。なお、この記憶は一時的なものである、(フラッシュメモリ 54 ではなく) プログラム用 RAM/ROM 53 の RAM 部分に記憶される。

#### 【0094】

ステップ S9 において、CPU 52 は、登録画像の中心である中心点  $C(i_n, j_n)$  を検出する。具体的には、図 12 に示されるように、画像 201 の縦軸の中心線と横軸の中心線が交差する点が登録画像の中心点とされる。ここで求める中心は、登録画像の中心であって、指紋の中心ではないので、迅速かつ簡単に求めることができる。なお、いまの例の場合、画像 200 は  $128 \text{ 画素} \times 128 \text{ 画素}$  とされているので、CPU 52 は、あらかじめ設定されている点  $(64, 64)$  (または、 $(63, 63)$ ) を中心点  $C$  として検出する。

#### 【0095】

ステップ S10 において、CPU 52 は、ステップ S9 の処理により求めた登録画像の中心である中心点  $C(i_n, j_n)$  と、ステップ S7 の処理により検出した複数の分岐点の距離をそれぞれ演算する。いまの例の場合、中心点  $C$  と分岐点  $P_1$  乃至  $P_8$  の間の距離がそれぞれ演算される。中心点  $C$  と分岐点  $P_1$  の間の距離を  $L_{p1}$  とすると、中心点  $C$  と分岐点  $P_2$  の間の距離は  $L_{p2}$  とされ、中心点  $C$  と分岐点  $P_3$  の間の距離は  $L_{p3}$  とされる。以下、順次、中心点  $C$  と  $P_4, P_5, \dots, P_8$  間の距離が、 $L_{p4}, L_{p5}, \dots, L_{p8}$  とされる。

#### 【0096】

ステップ S11 において、CPU 52 は、ステップ S10 の処理により演算した距離 (中心点  $C$  と分岐点  $P_1$  乃至  $P_8$  の間の距離) を短いものの順にソートする。いまの例の場合、中心からの距離は、 $L_{p1} < L_{p2} < L_{p3} < L_{p4} < L_{p5} < L_{p6} < L_{p7} < L_{p8}$  とされる。

#### 【0097】

ステップ S12 において、CPU 52 は、中心点  $C(i_n, j_n)$  と、ステップ S7 の処理により検出した複数の端点の距離をそれぞれ演算する。いまの例の場

合、中心点Cと端点Q1乃至Q10の間の距離がそれぞれ演算される。中心点Cと端点Q1間の距離を $L_{q1}$ とすると、中心点Cと端点Q2の間の距離は $L_{q2}$ とされ、中心点Cと端点Q3の間の距離は $L_{q3}$ とされる。以下、順次、中心点CとQ4, Q5, ..., Q10間の距離が $L_{q4}$ ,  $L_{q5}$ , ...,  $L_{q10}$ とされる。

#### 【0098】

ステップS13において、CPU52は、ステップS12の処理により演算した距離（中心点Cと端点Q1乃至Q10の間の距離）を短いものの順にソートする。いまの例の場合、中心からの距離は、 $L_{q1} < L_{q2} < L_{q3} < L_{q4} < L_{q5} < L_{q6} < L_{q7} < L_{q8} < L_{q9} < L_{q10}$ とされる。

#### 【0099】

ステップS14において、CPU52は、指紋の中心点C( $i_n, j_n$ )（ステップS9の処理により求めた中心点）に近い（いまの例の場合、最大10個までの）分岐点において、互いに最も近い3点を結ぶ3角形を全て作成する。なお、上述したように、いまの例の場合、CPU52は、指紋の中心点に近い最大10個までの分岐点において、互いに最も近い3点を結ぶ3角形を全て作成するとしているが、これに限らず、ステップS7の処理により検出された分岐点の個数と対応して、変更させることもできる。

#### 【0100】

具体的には、CPU52は、中心点C( $i_n, j_n$ )に最も近い分岐点P1( $x_{p1}, y_{p1}$ )（中心点Cからの距離 $L_{p1}$ が最も小さいため、分岐点P1が選択される）と、分岐点P1に最も近い2つの分岐点P2( $x_{p2}, y_{p2}$ )およびP3( $x_{p3}, y_{p3}$ )を結んで、図13に示されるような3角形W1を作成する。

#### 【0101】

その後、CPU52は、中心点Cに2番目に近い分岐点P2( $x_{p2}, y_{p2}$ )（中心点Cからの距離 $L_{p2}$ が2番目に小さいため、分岐点P2が選択される）と、分岐点P2に最も近い2つの分岐点P1( $x_{p1}, y_{p1}$ )および分岐点P3( $x_{p3}, y_{p3}$ )を結んで、3角形を作成する。いまの場合、上述した3角

形W1と全く同じである。以下、全く同じ3角形が作成される場合、その記載は省略する。

#### 【0102】

これを順次繰り返すことにより、中心点Cに3番目に近い分岐点P3と、分岐点P3に最も近い2つの分岐点P2, P4を結んだ3角形W2、中心点Cに4番目に近い分岐点P4と、分岐点P4に最も近い2つの分岐点P3, P7を結んだ3角形W3、および、中心点Cに5番目に近い分岐点P5と、分岐点P5に最も近い2つの分岐点P6, P8を結んだ3角形W4の4個の3角形が作成される（中心点Cに6番目に近い分岐点P6、中心点Cに7番目に近い分岐点P7、および中心点Cに8番目に近い分岐点P8に基づいて作成される3角形は、それぞれ、上述した3角形W1, W2, W3, またはW4のいずれかと重複するため、省略している）。

#### 【0103】

このように、中心から近い順番に、各分岐点と、それに最も近い2つの分岐点により3角形を作成すると、例えば、1つ目の3角形と2つ目の3角形、または3つ目の3角形が全く同じ分岐点により構成される場合も生じる。このことは、中心に近い分岐点から順番に、漏れることなく、各分岐点とそれに最も近い2つの分岐点において3角形を作成していることを示している。具体的には、中心点C( $i_n, j_n$ )に最も近い分岐点P1( $x_{p1}, y_{p1}$ )と、分岐点P1に最も近い2つの分岐点P2( $x_{p2}, y_{p2}$ )およびP3( $x_{p3}, y_{p3}$ )を結んで作成された3角形W1と、中心点Cに2番目に近い分岐点P2( $x_{p2}, y_{p2}$ )と、分岐点P2に最も近い2つの分岐点P1( $x_{p1}, y_{p1}$ )および分岐点P3( $x_{p3}, y_{p3}$ )を結んで作成された3角形は、3角形W1と全く同じであるが、このような3角形も、漏れることなく作成される。

#### 【0104】

ステップS15において、CPU52は、作成した複数の3角形の面積と辺の長さを計算する。いまの例の場合、作成された3角形は4個（3角形W1, W2, W3, およびW4）である（実際には、重複する3角形が作成されているため、4個以上の3角形が作成されているが、いまの例の場合では、異なる3角形は4

個とされる)ので、4個のそれぞれについて、面積と各辺の長さが計算される。これにより、3角形W1については、3角形W1の面積S1、分岐点P1と分岐点P2(以下、このような場合、P1P2のように記載する)間の距離、P2P3間の距離、およびP3P1間の距離が計算され、3角形W2については、3角形W2の面積S2、P2P3、P3P4、およびP4P2間の距離が計算され、3角形W3については、3角形W3の面積S3、P3P4間の距離、P4P7間の距離、およびP7P3間の距離が計算され、3角形W4については、3角形W4の面積S4、P5P6間の距離、P6P8間の距離、およびP8P5間の距離(辺の長さ)が求められる。

#### 【0105】

ステップS16において、CPU52は、フラッシュメモリ54に、それぞれの3角形の面積と各辺の長さを登録テンプレートに記憶させる。このように面積と辺の長さの集合に過ぎないので、そのデータ量は少ない。

#### 【0106】

ステップS17において、CPU52は、それぞれの3角形の3点に最も近い端点の位置を計算する。具体的には、図13の例の場合、3角形W1を構成する分岐点は、分岐点P1、P2およびP3とされるので、CPU52は、分岐点P1に最も近い端点Q2( $x_{q2}$ ,  $y_{q2}$ )、分岐点P2に最も近い端点Q1( $x_{q1}$ ,  $y_{q1}$ )、および分岐点P3に最も近い端点Q1( $x_{q1}$ ,  $y_{q1}$ )を検出し、それぞれの方向と距離(P1Q2間の方向と距離、P2Q1間の方向と距離、P3Q1間の方向と距離)を計算する。なお、いまの例の場合、位置として、方向と距離を求めるようにしたが、少なくともいずれか1つとしてもよい。この処理が繰り返され、3角形W2についてP3Q1間、P2Q1間、P4Q7間の方向と距離(以下、これを、位置とも称する)が求められ、3角形W3についてP3Q1間、P4Q7間、および、P7Q7間の位置が求められ、3角形W4についてP5Q6間、P6Q6間、およびP8Q6間の位置が求められる。

#### 【0107】

ステップS18において、CPU52は、ステップS17の処理により求めた端点の位置をそれぞれの3角形の面積と各辺の長さ(ステップS16の処理により

記憶した3角形の面積と各辺の長さ)に関連付し、登録テンプレートとして記憶させる。いまの例の場合、3角形W1, W2, W3およびW4の面積S1, S2, S3, S4、各辺(3辺)の長さ(例えば、W1の場合、P1P2間の長さ、P2P3間の長さ、P3P1間の長さ)、端点の位置(3角形W1の場合、P1Q2間の位置、P2Q1間の位置、P3Q1間の位置)がそれぞれ記憶される。なお、いまの例の場合、3角形W1においてP3Q1の位置が求められ、3角形W2においてもP3Q1の位置が求められる(同じ分岐点に対応する同じ端点との間の位置が求められる)ようになっているが、同じ点の位置については、計算を省略するようにしてもよい。

#### 【0108】

ステップS19において、CPU52は、USBコントローラ55を介して、図示せぬ外部の機器から供給された、指紋を読み取っている使用者のIDを取得し、フラッシュメモリ54に記憶されている登録テンプレート(ステップS16とステップS18の処理によりフラッシュメモリ54に記憶された登録テンプレート)に関連付けして記憶させる。

#### 【0109】

図6と図7の処理により、使用者のIDとともに、指紋画像から抽出した複数の3角形の面積、各辺の長さ、特定の分岐点(3角形を構成する分岐点)に対応する(に最も近接する)端点の位置(方向または距離)をその指紋の特徴として記憶させることができる。

#### 【0110】

なお、ステップS19の使用者のIDを記憶させる処理は、認証登録処理を開始するとき、予め(ステップS1の前に)行なわれていてもよい。

#### 【0111】

次に、図14乃至図17のフローチャートを参照して、図2の指紋照合装置1における指紋照合処理を説明する。なお、この処理は、図6と図7の指紋登録処理が行なわれた後に実行される処理であり、指紋読み取りセンサ21に、ユーザの指が押圧されたとき開始される。なお、ステップS101乃至ステップS113の処理は、上述した図6のステップS1乃至ステップS13と殆ど同様である

ので、簡単に説明する。

【0112】

ステップS101において、指読み取りセンサ21は、指紋読み取りセンサ21に押圧され、接触された指の指紋の画像を読み取り、読み取った指紋の画像に対応する信号を、指紋読み取り用LSI51に供給する。

【0113】

ステップS102において、指紋読み取り用LSI51は、指紋読み取りセンサ21から供給された指紋に対応する信号から、指紋の画像データを生成する。

【0114】

ステップS103において、指紋読み取り用LSI51は、生成した指紋の画像データを、256階調のグレー画像データに変換し、変換した256階調のグレー画像データをCPU52に供給する。

【0115】

ステップS104において、CPU52は、変換されたグレー画像データに対して、ノイズ除去や指紋隆線内の微少な傷を埋めるなどの前処理を実行する。具体的には、図18に示されるような、128画素×128画素のグレー画像300のデータが生成される。画像301は、96画素×96画素の画像である。図18に示されるグレー画像と、図9のグレー画像200を比較すると、図9の画像に比べて指紋の位置が右上にずれている。

【0116】

ステップS105において、CPU52は、前処理されたグレー画像データを、所定の閾値で2値化し、2値化画像データに変換する。図19は、2値化画像を表わしている。

【0117】

ステップS106において、CPU52は、細線化処理を行なう。これにより、図20に示されるような画像に変換される。

【0118】

ステップS107において、CPU52は、細線化処理された画像から、分岐点と端点をそれぞれ最大10個まで検出する。この処理により検出された分岐点と



端点は、図 2 1 に示されるようになる。このとき、検出される分岐点と端点は、実際には、図 2 1 の 128 画素×128 画素の画像 3 0 0 のうち、96 画素×96 画素で示される画像 3 0 1 が対象となる。図 2 1 において、分岐点は、P b 1 乃至 P b 7 (7 個) とされ、端点は、Q b 1 乃至 Q b 1 0 (10 個) とされる。

#### 【0119】

なお、いまの例の場合、CPU 5 2 は、検出する分岐点と端点を最大 10 個としたが、上述した図 6 および図 7 の処理と同様に、これに限らず、例えば、前もって決められた最大数のリミットとしてもよいし、10 個以上と設定してもよい。また、CPU 5 2 は、検出する分岐点と端点の対象を、96 画素×96 画素で示される画像 3 0 1 としたが、これに限らず、例えば、112 画素×112 画素で示される画像とすることもできる。

#### 【0120】

図 1 3 と図 2 1 を比較するに、図 1 3 の分岐点 P 1 は図 2 1 の分岐点 P b 1 に対応しており、分岐点 P 2 は分岐点 P b 2 に対応しており、分岐点 P 3 は分岐点 P b 5 に対応しており、分岐点 P 4 は分岐点 P b 7 に対応しており、分岐点 P 5 は分岐点 P b 3 に対応しており、分岐点 P 6 は分岐点 P b 4 に対応しており、分岐点 P 8 は分岐点 P b 6 に対応している。すなわち、図 1 3 と図 2 1 では、取り込まれた画像の位置が異なるため（ユーザが指を置く位置は、その都度変化するため）、図 1 3 において検出された分岐点 P 7 は、図 2 1 では検出されていない。また、分岐点 P b i の座標は、分岐点 P b i ( $x_{pb i}$ ,  $y_{pb i}$ ) (ただし、 $i = 1, 2, \dots, 7$ ) とされ、端点 Q b i の座標は端点 Q b i ( $x_{qb i}$ ,  $y_{qb i}$ ) (ただし、 $i = 1, 2, \dots, 10$ ) とされる。

#### 【0121】

ステップ S 1 0 8 において、CPU 5 2 は、プログラム用 RAM/ROM 5 3 の RAM 部分に、検出した分岐点と端点の座標 ( $x, y$ ) (すなわち、P b i ( $x_{pb i}$ ,  $y_{pb i}$ ) (ただし、 $i = 1, 2, \dots, 7$ )、および Q b i ( $x_{qb i}$ ,  $y_{qb i}$ ) (ただし、 $i = 1, 2, \dots, 10$ )) を記憶させる。なお、この記憶は一時的なものであるので、(フラッシュメモリ 5 4 ではなく) プログラム用 RA

M/ROM 53 のRAM部分に記憶される。

#### 【0122】

ステップ S109 において、CPU 52 は、指紋画像の中心点  $C_b(i_n, j_n)$  を検出する。なお、画像 300 は 128 画素  $\times$  128 画素とされているので、上述した図 6 のステップ S9 における場合と同様に、CPU 52 は、あらかじめ設定されている点 (64, 64) (または、(63, 63)) を、中心点として検出するようにしてもよい。

#### 【0123】

ステップ S110 において、CPU 52 は、ステップ S109 の処理により求めた指紋画像の中心点  $C_b(i_n, j_n)$  とステップ S107 の処理により検出した複数の分岐点の距離をそれぞれ演算する。いまの例の場合、中心点  $C_b$  と分岐点  $P_{b1}$  乃至  $P_{b7}$  の間の距離がそれぞれ演算される。中心点  $C_b$  と分岐点  $P_{b1}$  の間の距離を  $L_{pb1}$  とすると、中心点  $C_b$  と分岐点  $P_{b2}$  の間の距離は  $L_{pb2}$  とされ、中心点  $C_b$  と分岐点  $P_{b3}$  の間の距離は  $L_{pb3}$  とされる。以下、順次、中心点  $C_b$  と  $P_{b4}$ ,  $P_{b5}$ ,  $\dots$ ,  $P_{b7}$  間の距離が、 $L_{pb4}$ ,  $L_{pb5}$ ,  $\dots$ ,  $L_{pb7}$  とされる。

#### 【0124】

ステップ S111 において、CPU 52 は、ステップ S110 の処理により演算した距離 (中心点  $C_b$  と分岐点  $P_{b1}$  乃至  $P_{b7}$  の間の距離) を短いものの順にソートする。いまの例の場合、中心からの距離は、 $L_{pb1} < L_{pb2} < L_{pb3} < L_{pb4} < L_{pb5} < L_{pb6} < L_{pb7}$  とされる。

#### 【0125】

ステップ S112 において、CPU 52 は、中心点  $C_b(i_n, j_n)$  と、ステップ S107 の処理により検出した複数の端点の距離をそれぞれ演算する。いまの例の場合、中心点  $C_b$  と端点  $Q_{b1}$  乃至  $Q_{b10}$  の間の距離がそれぞれ演算される。中心点  $C_b$  と端点  $Q_{b1}$  間の距離を  $L_{qb1}$  とすると、中心点  $C_b$  と端点  $Q_{b2}$  の間の距離は  $L_{qb2}$  とされ、中心点  $C_b$  と端点  $Q_{b3}$  の間の距離は  $L_{qb3}$  とされる。以下、順次、中心点  $C_b$  と  $Q_{b4}$ ,  $Q_{b5}$ ,  $\dots$ ,  $Q_{b10}$  間の距離が  $L_{qb4}$ ,  $L_{qb5}$ ,  $\dots$ ,  $L_{qb10}$  とされる。

## 【0126】

ステップS113において、CPU52は、ステップS12の処理により演算した距離（中心点Cと端点Q1乃至Q8の間の距離）を短いものの順にソートする。いまの例の場合、中心からの距離は、 $L_{qb1} < L_{qb2} < L_{qb3} < L_{qb4} < L_{qb5} < L_{qb6} < L_{qb7} < L_{qb8} < L_{qb9} < L_{qb10}$ とされる。

## 【0127】

ステップS114において、CPU52は、指紋の中心点 $C_b(i_n, j_n)$ （ステップS109の処理により求めた中心点）に最も近い分岐点と、その分岐点から最も近い分岐点を2点検出し、その3点を結ぶ3角形を作成する。具体的には、図21の例の場合、CPU52は、中心点 $C_b$ に最も近い分岐点 $P_{b1}$ と、分岐点 $P_{b1}$ から最も近い分岐点 $P_{b2}$ 、および分岐点 $P_{b5}$ を結ぶ3角形 $W_{b1}$ を作成する。すなわち、CPU52は、中心点 $C_b(i_n, j_n)$ に最も近い分岐点 $P_{b1}(x_{pb1}, y_{pb1})$ （中心点 $C_b$ からの距離 $L_{pb1}$ が最も小さいため、分岐点 $P_{b1}$ が選択される）と、分岐点 $P_{b1}$ に最も近い2つの分岐点 $P_{b2}(x_{pb2}, y_{pb2})$ および分岐点 $P_{b5}(x_{pb5}, y_{pb5})$ の3点を結んで、3角形 $W_{b1}$ を作成する。

## 【0128】

なお、図21の例の場合、3角形 $W_1$ は3角形 $W_{b1}$ に対応しており、3角形 $W_2$ は3角形 $W_{b3}$ （このフローチャートでは作成されないが、3角形を全て作成した場合、3角形 $W_{b3}$ が作成される）に対応しており、3角形 $W_4$ は後述する3角形 $W_{b2}$ に対応している。すなわち、図13の3角形 $W_3$ は図21では作成されない。

## 【0129】

ステップS115において、CPU52は、ステップS114の処理により作成した3角形 $W_{b1}$ （分岐点 $P_{b1}$ 、 $P_{b2}$ および $P_{b5}$ からなる3角形 $W_{b1}$ ）の面積 $S_{b1}$ を計算する。

## 【0130】

ステップS116において、CPU52は、フラッシュメモリ54に記憶されて

いる登録テンプレートを読み出す。

#### 【0131】

ステップS117において、CPU52は、登録テンプレート（指紋登録処理（図6と図7の処理）によりフラッシュメモリ54に記憶された登録テンプレート）に、いま計算した3角形Wb1の面積Sb1と一致する面積を有する登録テンプレートがあるか否かを判定する。指紋は、周辺環境、または体調などによりある程度変形するため、CPU52は、許容範囲を設け、完全に一致しない場合においても、許容範囲内の（すなわち、照合閾値より小さい）値であれば、一致するように判定する。いまの例の場合、図21の3角形Wb1は、図7のステップS15の処理において求められた図13の3角形W1と対応しているので、その面積Sb1とS1は一致すると判定される。

#### 【0132】

ステップS117において、3角形Wb1の面積Sb1と一致する（面積が一致する3角形が、登録テンプレートにある）と判定された場合、ステップS118において、CPU52は、3角形Wb1の辺の長さを計算する。具体的には、Pb1Pb2間の長さ、Pb2Pb5間の長さ、Pb5Pb1間の長さが計算される。なお、この長さは、ステップS114において、面積と同時に計算するようにしてもよい。

#### 【0133】

ステップS119において、CPU52は、ステップS117の処理により一致すると判定された、登録テンプレートに登録されている3角形（いまの例の場合、3角形W1）を構成する3辺の長さと、ステップS118の処理により計算した3辺の長さ（いまの例の場合、3角形Wb1の3辺の長さ）が一致するか否かを判定する。いまの例の場合、3角形W1に対応するP1P2間の長さ、P2P3間の長さ、およびP3P1間の長さ（図7のステップS16の処理により記憶された辺の長さ）と、ステップS118の処理により計算した3辺の長さPb1Pb2間、Pb2Pb5間、およびPb5Pb1間の長さが一致するか否かが判定される。勿論、CPU52は、許容範囲を設け、完全に一致しない場合においても、許容範囲内の（すなわち、照合閾値より小さい）値であれば、一致するよう

に判定する。3 角形  $W1$  と 3 角形  $Wb1$  を構成するそれぞれの分岐点  $P1$  と分岐点  $Pb1$ 、分岐点  $P2$  と分岐点  $Pb2$ 、および分岐点  $P3$  と分岐点  $Pb5$  は対応しているので、勿論、一致すると判定される。

#### 【0134】

ステップ  $S119$  において、3 辺の長さが一致すると判定された場合、ステップ  $S120$  において、CPU52 は、その 3 角形（いまの場合、ステップ  $S114$  の処理により作成した 3 角形  $Wb1$ ）を構成する 3 個の分岐点（分岐点  $Pb1$ 、分岐点  $Pb2$ 、および分岐点  $Pb5$ ）のそれぞれに最も近い端点を検索する。いまの例の場合、分岐点  $Pb1$  に最も近い端点  $Qb3$  ( $xqb3$ ,  $yqb3$ )、分岐点  $Pb2$  に最も近い端点  $Qb7$  ( $xqb7$ ,  $yqb7$ )、および分岐点  $Pb5$  に最も近い端点  $Qb7$  ( $xqb7$ ,  $yqb7$ ) が検索される。

#### 【0135】

ステップ  $S121$  において、CPU52 は、ステップ  $S120$  の処理により検索した端点の位置を計算する。具体的には、 $Pb1Qb3$  間の位置、 $Pb2Qb7$  間の位置、 $Pb5Qb7$  間の位置が計算される。

#### 【0136】

ステップ  $S118$  の辺の長さの計算とステップ  $S120$ 、 $S121$  の端点の検索と位置の計算は、ステップ  $S115$  において、面積の計算時にまとめて行なうこともできる。ただし、面積が一致しなければ、後の 2 つの計算は不要となり、辺の長さが一致しなければ、端点の検索と位置の計算は不要となるので、前の条件の判定後、順次計算した方が、計算量が少なく済み、結果的に迅速な判定が可能になる。

#### 【0137】

ステップ  $S122$  において、CPU52 は、ステップ  $S117$  の処理により一致すると判定された 3 角形を構成する 3 点の分岐点と、それぞれの分岐点に最も近い端点との位置と、ステップ  $S121$  の処理により計算した  $Pb1Qb3$  間の位置、 $Pb2Qb7$  間の位置、および、 $Pb5Qb7$  間の位置を比較する。いまの例の場合、3 角形  $W1$  を構成する 3 点の分岐点  $P1$ 、 $P2$  および  $P3$  と、それぞれの分岐点に最も近い端点との位置、すなわち、図 7 のステップ  $S18$  の処理に

より記憶された P 1 Q 2 間の位置、P 2 Q 1 間の位置、および、P 3 Q 1 間の位置) と、ステップ S 1 2 1 の処理により計算した P b 1 Q b 3 間の位置、P b 2 Q b 7 間の位置、および、P b 5 Q b 7 間の位置が比較される。

#### 【0138】

ステップ S 1 2 3 において、CPU 5 2 は、ステップ S 1 2 2 の処理により比較した、3 角形を構成する 3 個の分岐点と、3 個の分岐点にそれぞれ最も近い端点との位置が一致するか否かを判定する。いまの例の場合、P 1 Q 2 間、P 2 Q 1 間、および P 3 Q 1 間の位置と、P b 1 Q b 3 間の位置、P b 2 Q b 7 間の位置、および、P b 5 Q b 7 間の位置が一致するか否かが判定される。3 角形 W b 1 により求められる P b 1 Q b 3、P b 2 Q b 7、および P b 5 Q b 7 は、それぞれ、3 角形 W 1 により求められる P 1 Q 2、P 2 Q 1、および P 3 Q 1 に対応しているので、一致すると判定される。なお、勿論、所定の範囲内の（すなわち、照合閾値より小さい）誤差は、一致すると判定される。一致すると判定された場合、処理はステップ S 1 2 4 に進み、CPU 5 2 は、カウンタに 1 を加える（1 だけインクリメントされる）。いまの例の場合、カウンタの値（初期値は 0 とされている）は 1 とされる。

#### 【0139】

ステップ S 1 2 5 において、CPU 5 2 は、カウンタの値が 2 であるか否かを判定する。カウンタの値は、作成された 1 つの 3 角形の面積、辺の長さ、それぞれの分岐点と端点との間の位置が全て一致した場合に加えられる。いまの例の場合、カウンタの値は 1 とされる（1 つの 3 角形について認証ができたとされる）ので、ステップ S 1 2 6 において、CPU 5 2 は、次の 3 角形が作成できるか否かを判定する。すなわち、次に中心点 C b に近い分岐点と、その分岐点に最も近い 2 つの分岐点を結んで、いままでに作成した 3 角形と異なる 3 角形が作成できるか否かが判定される。いまの例の場合、2 番目（分岐点 P 1 の次）に中心点 C b に近い分岐点は P 2 であり、分岐点 P 2 により作成される 3 角形は、3 角形 W b 1 と同じであるため新しい 3 角形は作成できないが、3 番目に中心点 C b に近い分岐点 P 3 を用いて作成する 3 角形 W b 2 が作成できるため、次の 3 角形は作成できると判定される。

## 【0140】

ステップS126において、次の3角形が作成できると判定された場合、ステップS127に進み、CPU52は、次の3角形を作成する。すなわち、ステップS107の処理により、図21から分岐点Pb1乃至Pb7と端点Qb1乃至Qb10が検出されているので、CPU52は、図6のステップS14の処理と同様にして、3番目に中心点Cbに距離が近い分岐点Pb3と、分岐点Pb3に最も近い2個の分岐点Pb4、および分岐点Pb6を結ぶ3角形Wb2を作成する。上述したように、ステップS126の処理では、2番目に中心点Cbに距離が近い分岐点Pb2により構成される3角形も作成されるが、分岐点Pb2と、分岐点Pb2に最も近い2点の分岐点は、分岐点Pb1とPb5とされ、3角形Wb1と同じとなるので、その後、3番目に中心点Cbに近い分岐点Pb3により構成される3角形が作成される。

## 【0141】

その後、処理はステップS115に戻り、CPU52は、ステップS127の処理により作成した3角形（いまの例の場合、Wb2）の面積を計算する。以下、同様の処理が繰り返され、いまの例の場合、3角形Wb2と登録テンプレートに登録されている図13の3角形W4が対応しているので、ステップS117、ステップS119、およびステップS123の処理は、全て、YES（一致する）と判定される。

## 【0142】

その結果、ステップS124において、カウンタに1が加えられる。いまの例の場合、カウンタの値は2とされる。

## 【0143】

ステップS125において、CPU52は、カウンタの値が2であるか否かを判定する。いまの例の場合、カウンタの値は2である（3角形Wb1が3角形W1と一致し、3角形Wb2と3角形W4が一致するため）と判定される。カウンタの値が2ということは、指紋画像の分岐点により作成される異なる2つの3角形において、3角形の面積、辺の長さ、それぞれの分岐点と端点との間の位置が登録テンプレートと全て一致していることを示している。カウンタの値が2である

と判定された場合、ステップ S 1 2 9 に進み、CPU 5 2 は、認証できた（すなわち、登録した使用者といま照合している使用者が一致する）ことを表わす信号を出力し、処理を終了する。照合の結果は、本体部 1 1 のランプ 2 2 とランプ 2 3 により点灯される。

#### 【0144】

ステップ S 1 1 7 において面積が一致しないと判定された場合、ステップ S 1 1 9 において長さが一致しないと判定された場合、ステップ S 1 2 3 において端点の位置が一致しないと判定された場合、または、ステップ S 1 2 5 においてカウンタの値が 2 ではないと判定された場合（すなわち、1 つの 3 角形について、3 角形の面積、辺の長さ、およびそれぞれの分岐点と端点との間の位置のうち、少なくともいずれか 1 つが一致しないと判定された場合）、ステップ S 1 2 6 に進み、次の 3 角形が作成できるか否かが判定される。次の 3 角形が作成できないことは、ステップ S 1 0 7 の処理により検出された分岐点 P b 1 乃至 P b 7 を全て使用して、過去に作成した 3 角形と重複しない 3 角形を作成し終えたことを意味する。次の 3 角形が作成できないと判定された場合、ステップ S 1 2 8 に進み、CPU 5 2 は、認証できなかった（すなわち、登録した使用者といま照合している使用者が一致しない）ことを表わす信号を出力し、処理を終了する。照合の結果は、本体部 1 1 のランプ 2 2 とランプ 2 3 により点灯される。

#### 【0145】

図 1 4 乃至図 1 7 の処理により、分岐点により求められる 3 角形の面積、3 辺の長さ、3 個の分岐点に最も近接するそれぞれの端点の位置により、指紋の照合を行なうことができる。また、取得した指紋画像の中心を求めて、中心に近い分岐点から 3 角形を順次生成することにより、一致する登録テンプレートを早く検索することができる。

#### 【0146】

ユーザは、指を指紋読み取りセンサ 2 1 上に置くとき、指の中心を指紋読み取り用センサ 2 1 の中心に合わせようとする。そして、指の向きはその都度異なるので、中心から遠い位置の画像程、登録画像と異なる画像となる可能性が高くなる。従って、一致する画像であれば、中心から判定を進めていくことにより、一



致するとの判定結果を早く得ることができる。

#### 【0147】

また、異なる3角形の場合、辺の長さと端点の位置に較べて、面積が一致する可能性の方が低い。従って、面積を先に判定した方が、無駄な判定の回数が減り、迅速な判断ができる。

#### 【0148】

2つの3角形において全て一致する場合（ステップS125においてYESと判定される場合）に、認証できたことにするのは（全ての3角形において一致する場合に、認証できたことにしないのは）、指紋読み取り用センサ21に押圧され、取得される指紋画像に、最大約3ミリのずれが生じるためである（指を置く毎に変化し易いためである）。すなわち、指紋画像にずれが生じることにより、作成される3角形が一部異なるためである（例えば、図21の例の場合、図13の3角形W3に対応する3角形は作成されない）。

#### 【0149】

例えば、図14のステップS104の処理により得られた、前処理されたグレー画像が、図22に示されるような画像である場合（勿論、同一の指から取得された指紋画像データとされる）、図9の登録画像（図6のステップS4の処理により取得された前処理されたグレー画像）と比較して、大幅に左下にずれている。

#### 【0150】

図22に示されるグレー画像は、2値化処理（ステップS105の処理）により、図23に示されるように変換され、さらに、細線化処理（ステップS106の処理）により、図24に示されるように変換される。そして、分岐点と端点の検出（ステップS107の処理）により、図25に示されるようになる。図25においては、図21の場合と同様に分岐点をそれぞれPb1乃至Pb7（図25の例の場合は、分岐点は7個しか検出されない）、端点がそれぞれQb1乃至Qb10、中心点がCbとそれぞれ設定される。

#### 【0151】

図13と図25を比較するに、図13の分岐点P1は図25の分岐点Pb3に

対応しており、分岐点P 2は分岐点P b 2に対応しており、分岐点P 3は分岐点P b 1に対応しており、分岐点P 4は分岐点P b 6に対応しており、分岐点P 7は分岐点P b 5に対応している。すなわち、図1 3と図2 5では、取り込まれた画像の位置が異なるため、図1 3において検出された分岐点P 5、分岐点P 6および分岐点P 8は、図2 5において検出されない。その代わり、図2 5においては、図1 3において検出されなかった分岐点である分岐点P b 4および分岐点P b 7が検出される。

#### 【0152】

CPU 5 2は、指紋画像の中心点C bを求めた後（ステップS 1 0 9の処理の後）、または、予め設定されている中心点C bを読み出した後、中心点C bと分岐点P b 1乃至P b 7の距離を短いものの順にソートし（ステップS 1 1 1）、さらに、中心点C bと端点Q b 1乃至Q b 1 0の距離を短いものの順にソートし（ステップS 1 1 3）、3角形を作成する（ステップS 1 1 4の処理）。図2 5の例の場合、ステップS 1 1 4またはステップS 1 2 7により、2個の3角形（3角形W b 1と3角形W b 2）が順次作成される。もし、順次、作成可能な全ての3角形が作成されたとしたら、図2 5の例の場合、作成される3角形は、順番に、分岐点P b 1、分岐点P b 2、および分岐点P b 6の3個を結ぶ3角形W b 1、分岐点P b 2、分岐点P b 1、および分岐点P b 3の3個を結ぶ3角形W b 2、分岐点P b 4、分岐点P b 5、および分岐点P b 7の3個を結ぶ3角形W b 3、分岐点P b 6、分岐点P b 1、および分岐点P b 5の3個を結ぶW b 4、並びに、分岐点p b 7、分岐点P b 5、および分岐点P b 6の3個を結ぶW b 5の5個の3角形が作成される（勿論、この5個の3角形と重複する3角形も作成されている）。すなわち、図2 1の3角形W b 3と3角形W b 5に対応する3角形は、図1 3に示される指紋画像では作成されない（すなわち、登録テンプレートにも登録されていないため、3角形W b 3とW b 5においては、一致しないと判定される）。

#### 【0153】

そこで、図1 4乃至2 0に示されるように、全てではなく、そのうちの複数個としての2個の3角形が一致する場合に、認証できたと判定するような処理にす

ることにより、正確に照合することができる。逆に、その数を1個としてしまうと、第3者の指が誤認証される恐れが高くなる。

#### 【0154】

登録された指紋画像が本人のものであるにもかかわらず、正しく照合されない確率をFRR (False Rejection Rate: 本人拒否率) といい、登録された指紋画像が任意の他人のものであっても、照合されてしまう確率をFAR (False Acceptance Rate; 他人受け入れ率) という。FARが高い指紋照合装置は、他人が本人に成りすますことが容易となり、セキュリティの質が低い指紋照合装置である。

#### 【0155】

照合閾値を高くする（一致するとみなす誤差の範囲を大きくしたり、一致する3角形の数の基準値（2個）をもっと大きくする）ことにより、FARを下げる事が可能となる。しかしながら、その場合、照合の可否は、照合時の環境（例えば、取得した指紋画像に混入されてしまうノイズなど）の変化の影響などを強く受けてしまい、本人であるにもかかわらず照合されない場合が増えてしまう（すなわち、FRRが高くなる）ため、ユーザの利便性が失われてしまう。そこで、照合閾値は、FRRとFARの両方のバランスを図る値に設定する必要がある。

#### 【0156】

以上の処理により、指紋画像の分岐点と端点を検出し、指紋画像の中心点からの距離に基づいて、分岐点（または端点）から近接する3個を結ぶ3角形を作成し、作成した3角形の面積、3辺の長さ、および3つの分岐点にそれぞれ最も近い端点との位置（距離または方向）を登録することにより指紋の照合を行なうようにしたので、保存する情報量（登録テンプレートサイズ）を従来より大幅に小さくすることができ、もって、メモリの容量を減らすことが可能となる。

#### 【0157】

また、簡単な登録データ（3角形の面積、3辺の長さ、および3つの分岐点にそれぞれ最も近い端点との位置）を用いるため、照合画像の位置や回転補正をする必要もなく、簡単な処理により、容易に照合を行なうことができ、もって処理速度を速くすることができる。

#### 【0158】

さらに、指紋照合の精度を向上させることができる。

#### 【0159】

また、中心点との距離に基づいて、分岐点と端点をソートし、そのソートした結果から順番に、分岐点または端点を検索するようにしたので、最も近い他の分岐点や端点を迅速に特定することができる。

#### 【0160】

なお、以上の例においては、3個の分岐点の最も近いそれぞれの端点の距離と方向を位置として登録し、指紋の照合をするようにしたが、距離と方向の少なくとも一方でよい。

#### 【0161】

なお、指紋画像の分岐点および端点の位置は、取得された指紋画像の左上からソートするようにしてもよいが、指紋読み取り用センサ21に押圧され、取得される指紋画像は、最大約3ミリのずれが生じる（指を置く毎に変化し易い）。このため、指紋画像の中心点を求め、中心点に対する分岐点と端点の位置を求めることにより一致する登録テンプレートを検索し易くすることが可能となる。

#### 【0162】

なお、以上の例においては、3つの分岐点により3角形を作成するようにしたが、端点により3角形を作成するようにしてもよい。この場合、端点により3角形が作成され、3角形を構成する辺の距離、および3角形を構成する3つの端点にそれぞれ最も近い分岐点との間の位置が求められる。

#### 【0163】

なお、以上の例においては、3つの分岐点により3角形を作成し、記憶するようにしたが、これに限らず、4角形以上の多角形とすることも可能である。ただし、3角形にする場合の方が、FARとFRRを低くすることができ、最も正確、かつ迅速な判定が可能となる。

#### 【0164】

上述した一連の処理は、ハードウェアにより実行させることもできるし、ソフトウェアにより実行させることもできる。一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、専用のハードウェ

アに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することが可能な、例えば汎用のパーソナルコンピュータなどに、記録媒体からインストールされる。

#### 【0165】

この記録媒体は、図3に示すように、コンピュータとは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク71（フレキシブルディスクを含む）、光ディスク72（CD-ROM(Compact Disc-Read Only Memory)、DVD(Digital Versatile Disc)を含む）、光磁気ディスク73（MD(Mini-Disc)（商標）を含む）、若しくは半導体メモリ74などよりなるパッケージメディアにより構成されるだけでなく、コンピュータに予め組み込まれた状態でユーザに提供される、プログラムが記録されているプログラム用RAM/ROM53や、ハードディスクなどで構成される。

#### 【0166】

なお、上述した一連の処理を実行させるプログラムは、必要に応じてルータ、モデムなどのインタフェースを介して、ローカルエリアネットワーク、インターネット、デジタル衛星放送といった、有線または無線の通信媒体を介してコンピュータにインストールされるようにしてもよい。

#### 【0167】

また、本明細書において、記録媒体に格納されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

#### 【0168】

##### 【発明の効果】

以上の如く、第1の本発明によれば、指紋照合に使用される情報量を小さくすることができる。記憶装置を小型化し、低コスト化することができる。従って、小型の装置への適用が可能となる。

#### 【0169】

第2の本発明によれば、指紋照合を行なうことができる。特に、小さい情報量

で指紋照合することができる。また、指紋照合の処理速度を速くすることができる。さらに、指紋照合の精度を向上させることができる。

【図面の簡単な説明】

【図 1】

本発明を適用した指紋照合装置の一実施の形態の外観の構成を示す斜視図である。

【図 2】

指紋読み取りセンサが、指紋が読み取られる指で押圧されている状態を説明する平面図である。

【図 3】

指紋照合装置の構成を示すブロック図である。

【図 4】

指紋読み取りセンサの原理的な構成を示す断面図である。

【図 5】

指紋読み取りセンサの内部回路の構成を示す回路図である。

【図 6】

図 3 の指紋照合装置における指紋登録処理を説明するフローチャートである。

【図 7】

図 3 の指紋照合装置における指紋登録処理を説明するフローチャートである。

【図 8】

図 6 のステップ S 3 における画像例を示す図である。

【図 9】

図 6 のステップ S 4 における画像例を示す図である。

【図 10】

図 6 のステップ S 5 における画像例を示す図である。

【図 11】

図 6 のステップ S 6 における画像例を示す図である。

【図 12】

図 6 のステップ S 7 における画像例を示す図である。

## 【図 13】

図 7 のステップ S 14 における画像例を示す図である。

## 【図 14】

図 3 の指紋照合装置における指紋照合処理を説明するフローチャートである。

## 【図 15】

図 3 の指紋照合装置における指紋照合処理を説明するフローチャートである。

## 【図 16】

図 3 の指紋照合装置における指紋照合処理を説明するフローチャートである。

## 【図 17】

図 3 の指紋照合装置における指紋照合処理を説明するフローチャートである。

## 【図 18】

図 14 のステップ S 104 における画像例を示す図である。

## 【図 19】

図 14 のステップ S 105 における画像例を示す図である。

## 【図 20】

図 14 のステップ S 106 における画像例を示す図である。

## 【図 21】

図 14 のステップ S 107 における画像例を示す図である。

## 【図 22】

図 14 のステップ S 104 における他の画像例を示す図である。

## 【図 23】

図 14 のステップ S 105 における他の画像例を示す図である。

## 【図 24】

図 14 のステップ S 106 における他の画像例を示す図である。

## 【図 25】

図 14 のステップ S 107 における他の画像例を示す図である。

## 【符号の説明】

1 指紋照合装置, 11 本体部, 21 指紋読み取りセンサ, 51  
指紋読み取り用LSI, 52 CPU, 53 プログラム用RAM/ROM, 54 フ

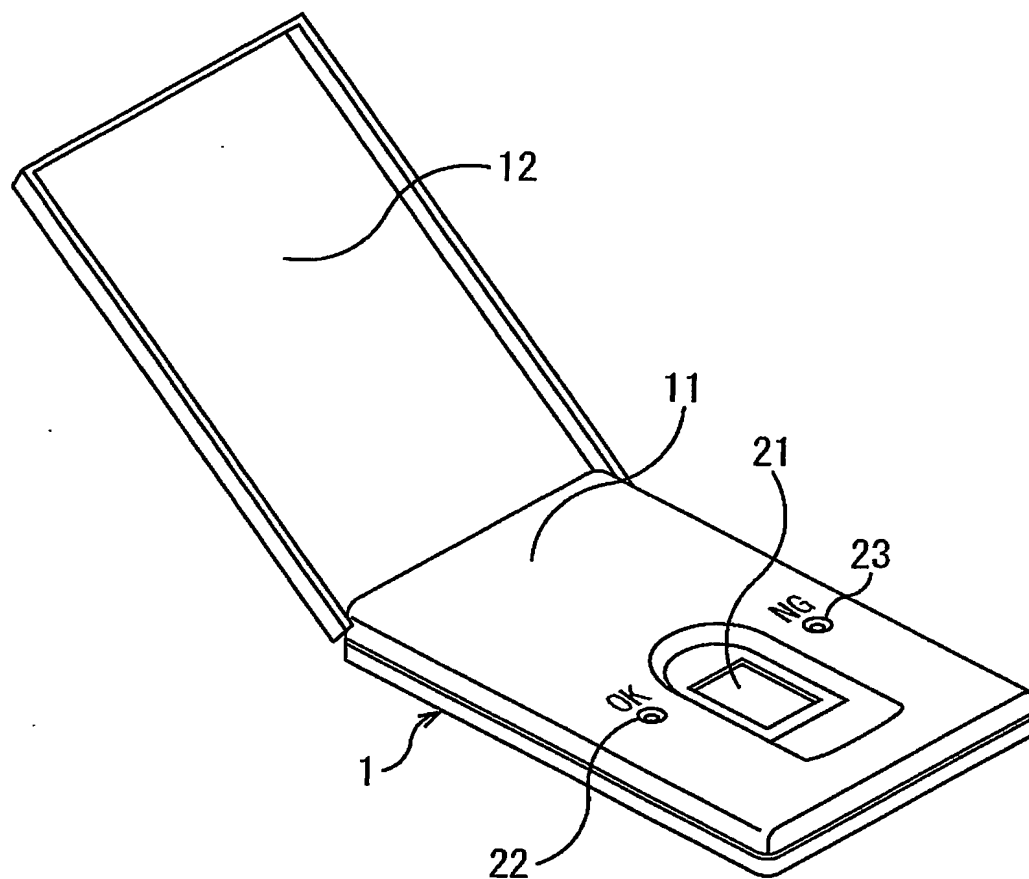
ラッシュメモリ, 7 1 磁気ディスク, 7 2 光ディスク, 7 3 光磁気  
ディスク, 7 4 半導体メモリ



【書類名】 図面

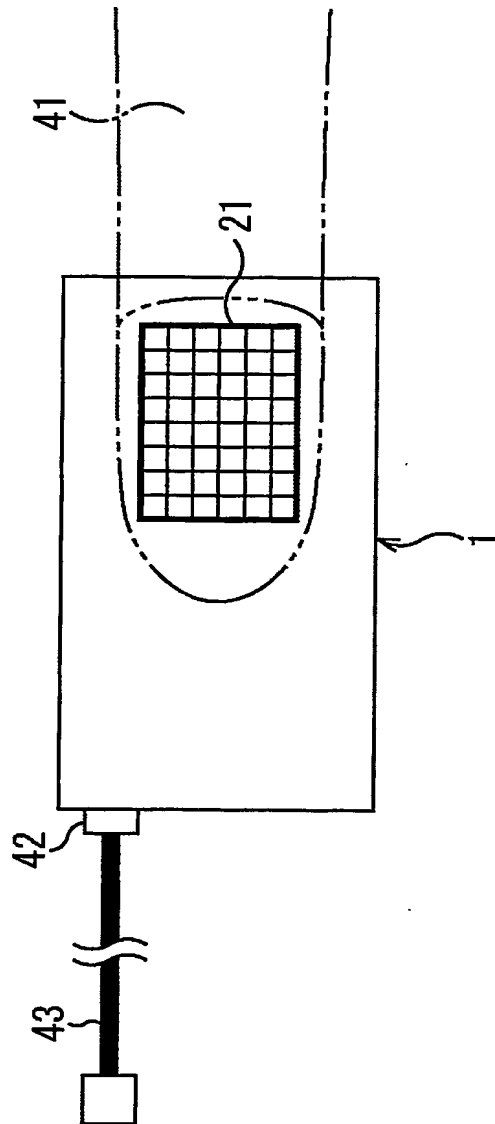
【図 1】

図1



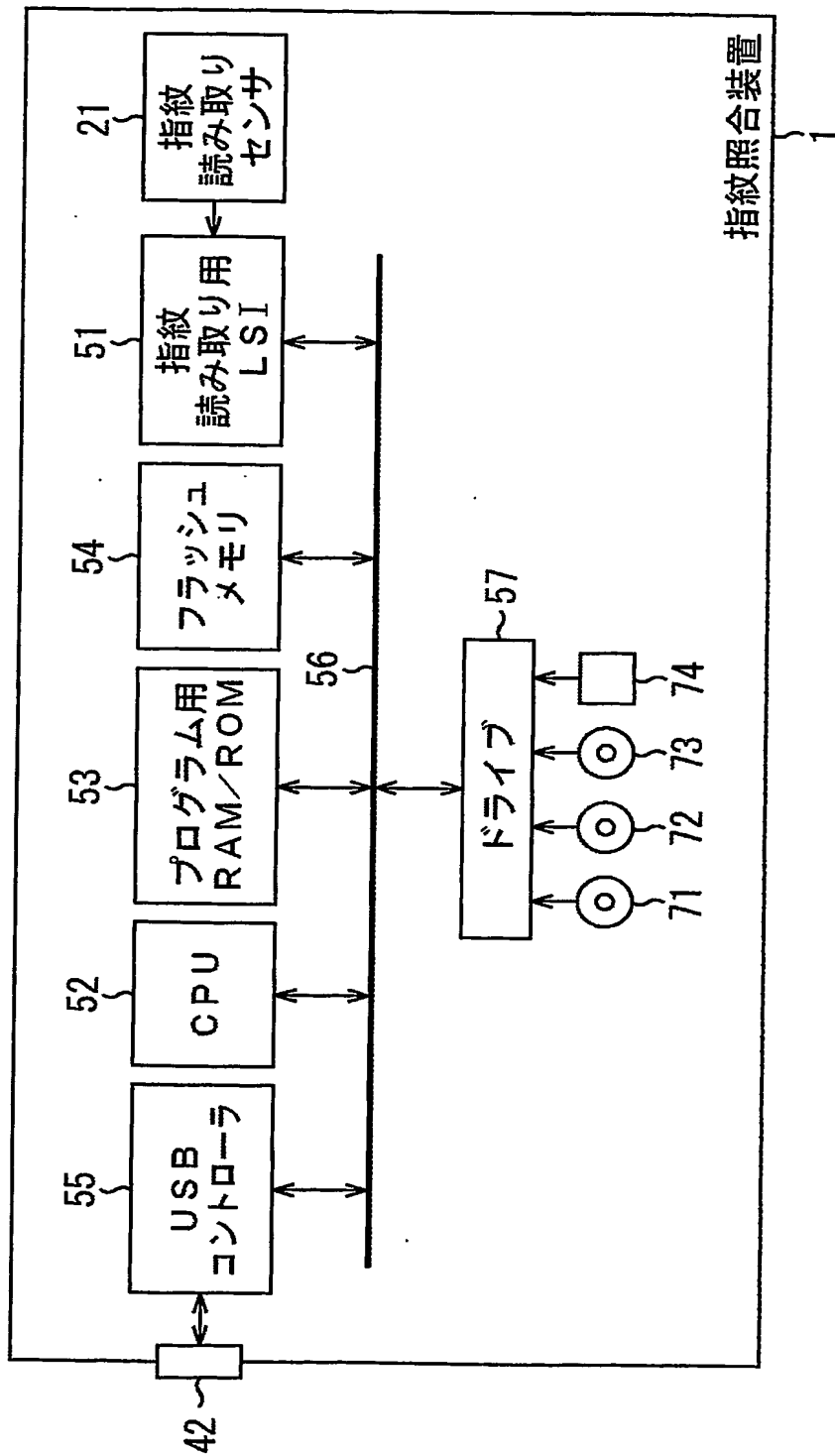
【図 2】

図2



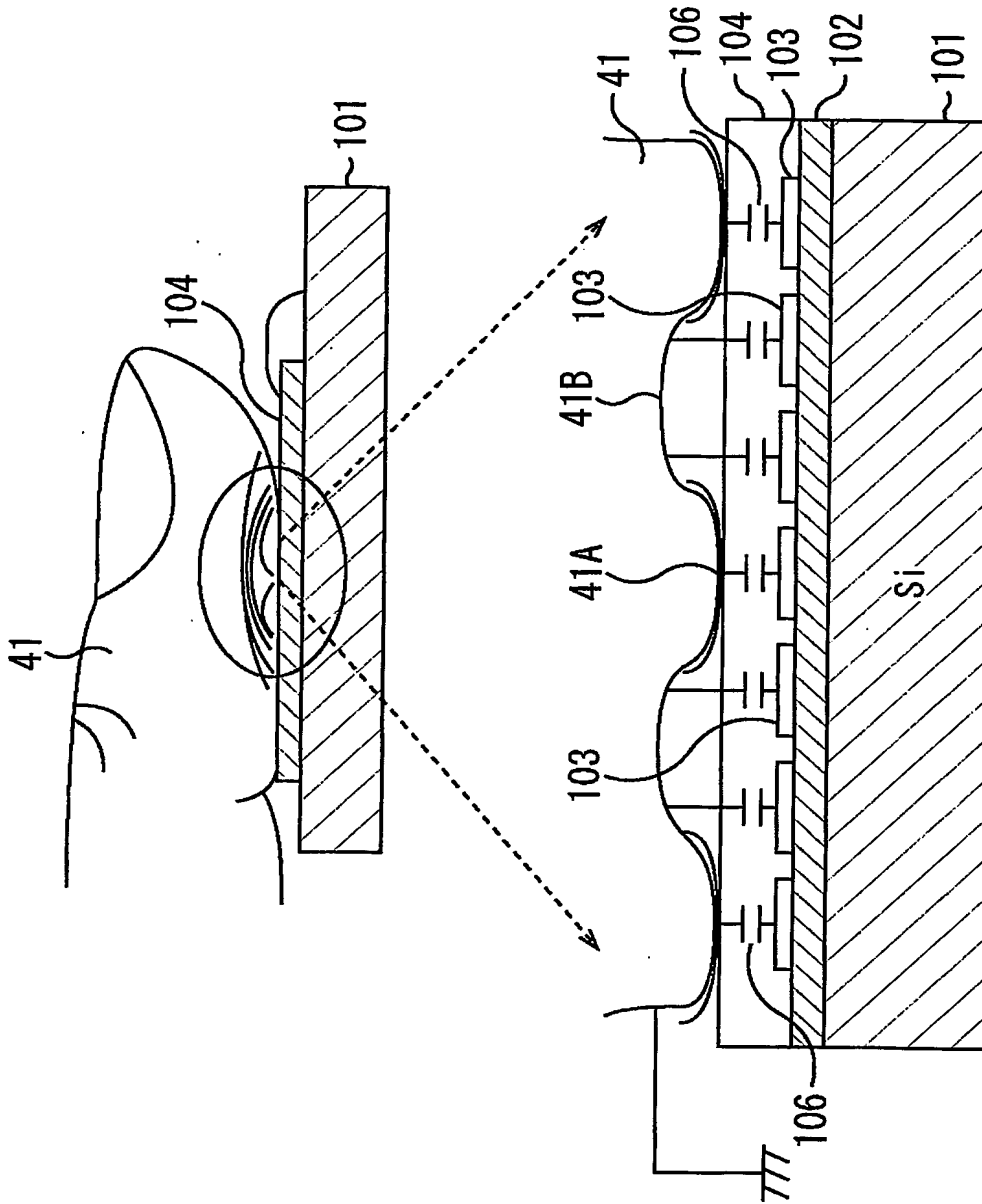
【図 3】

図3



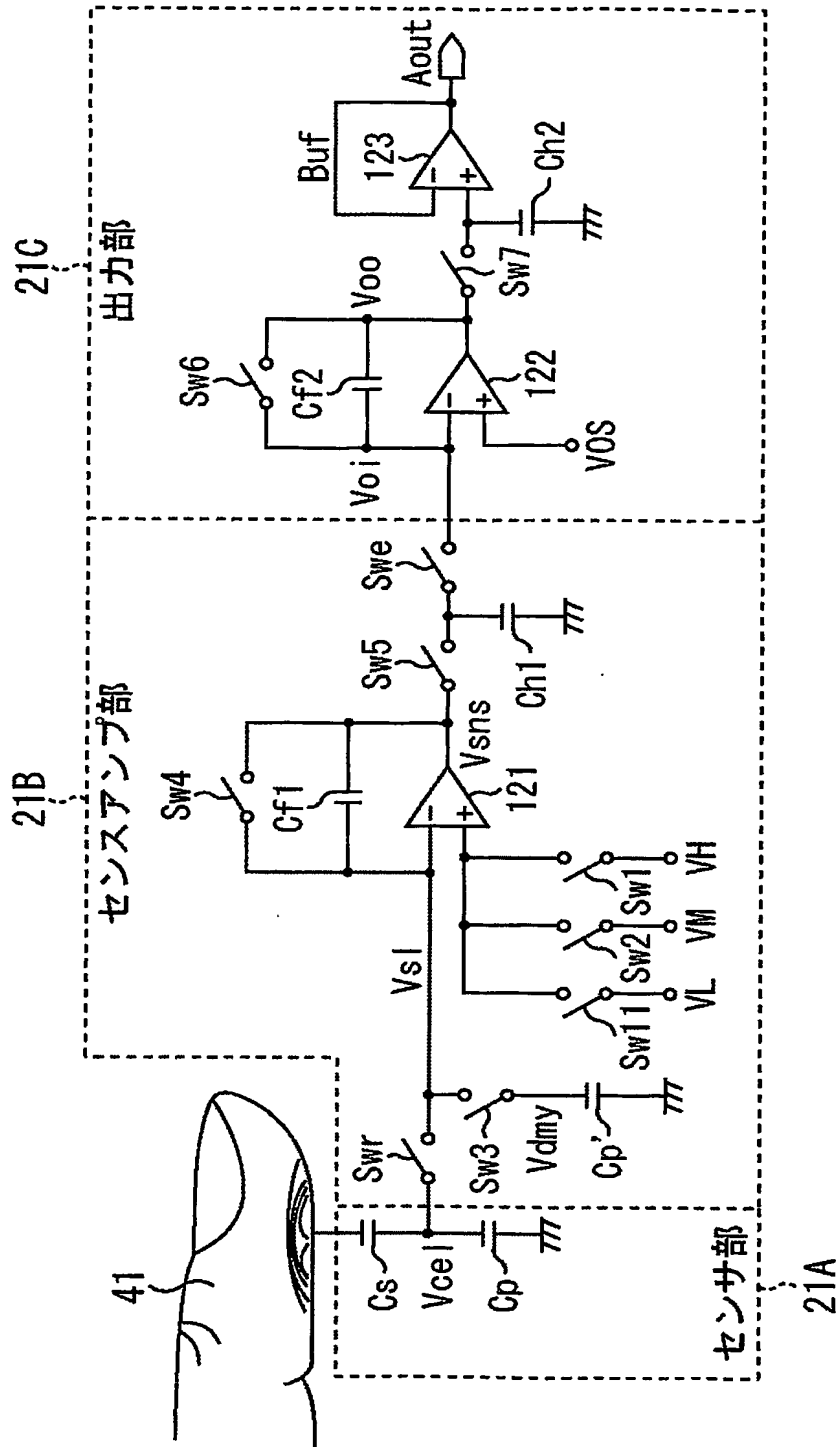
【図 4】

図4



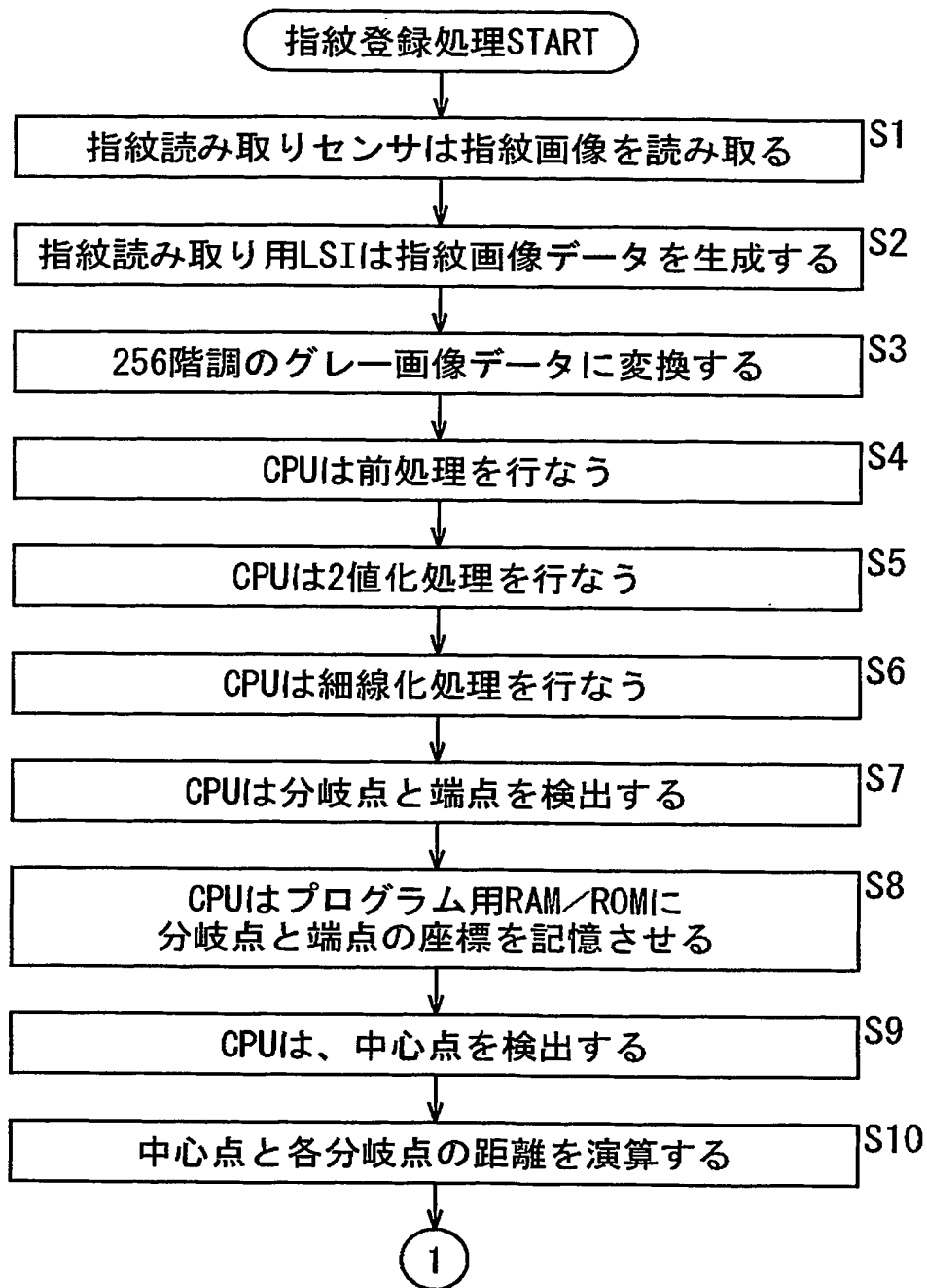
【図 5】

図5



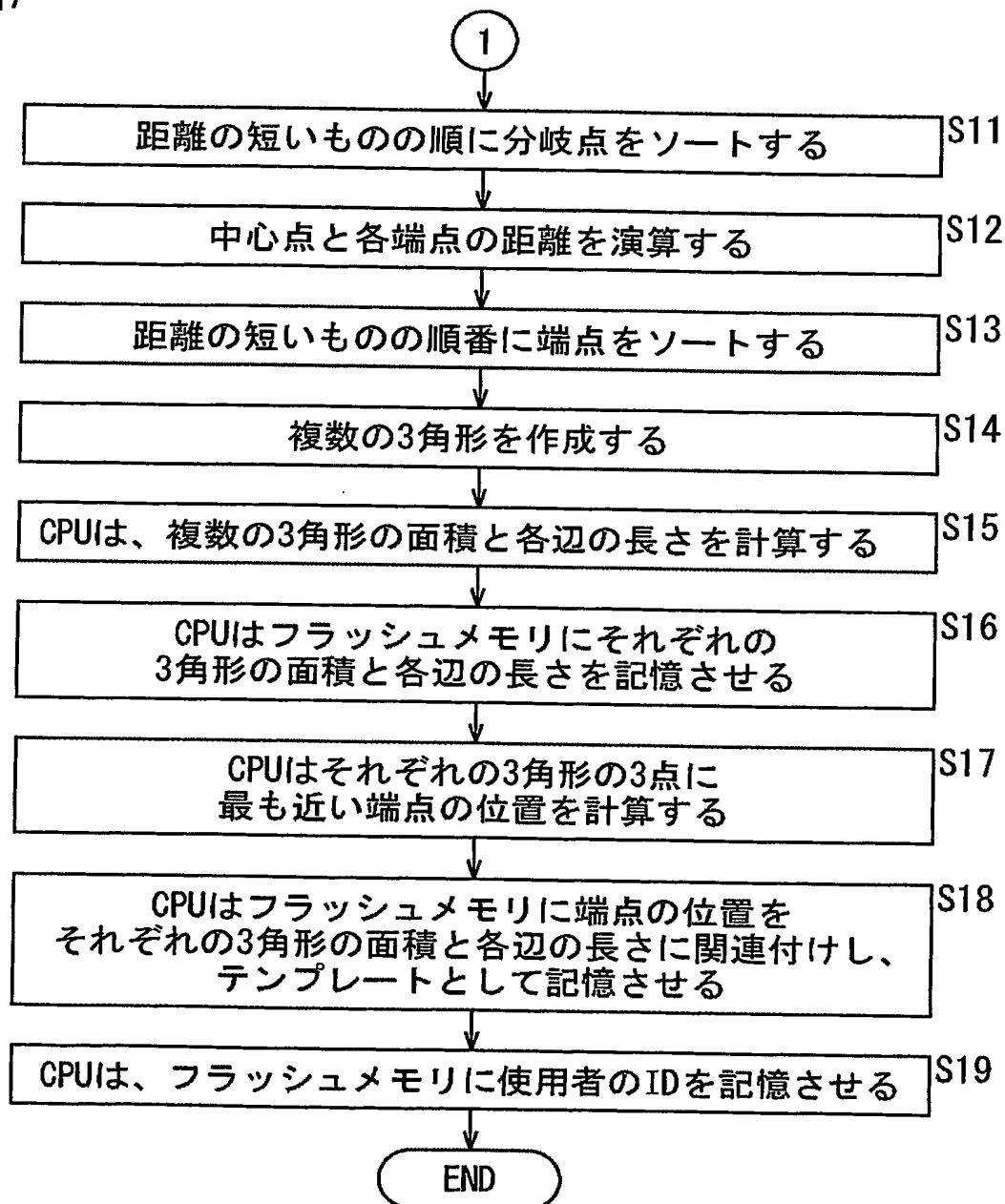
【図6】

図6



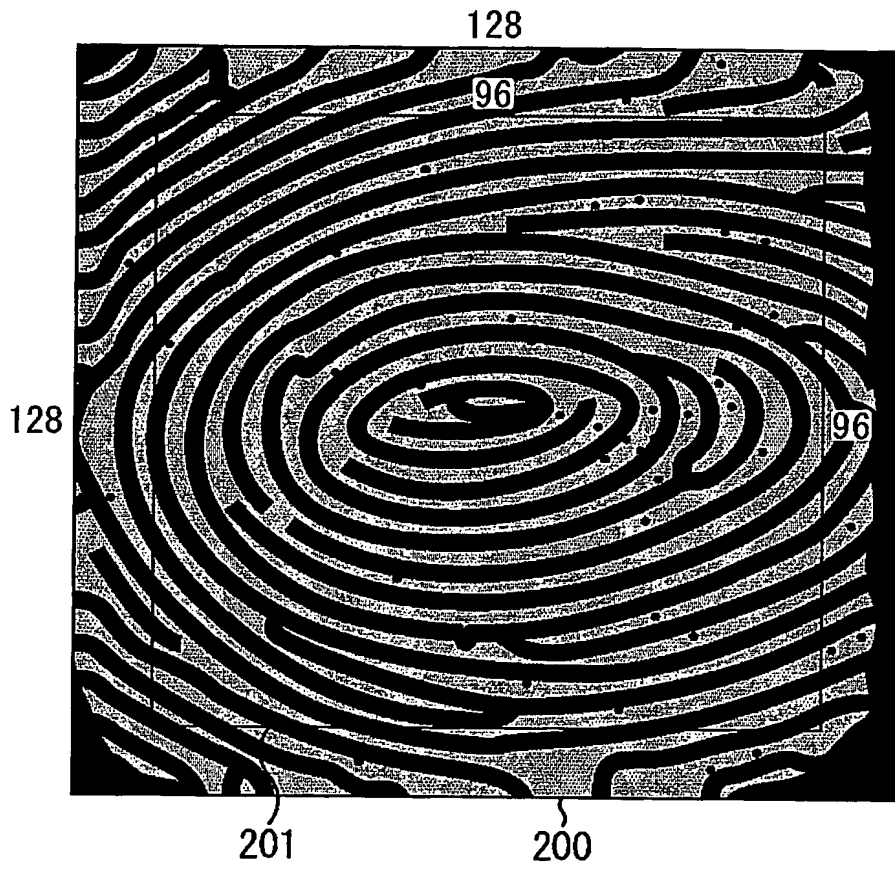
【図 7】

図 7



【図 8】

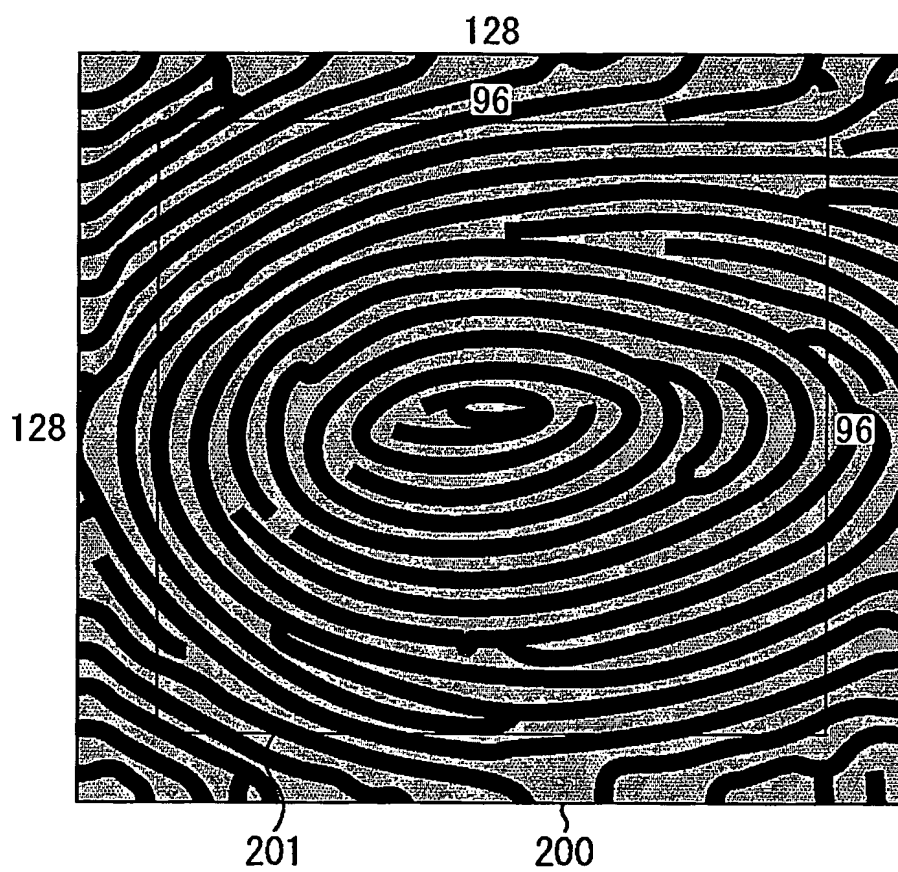
図 8





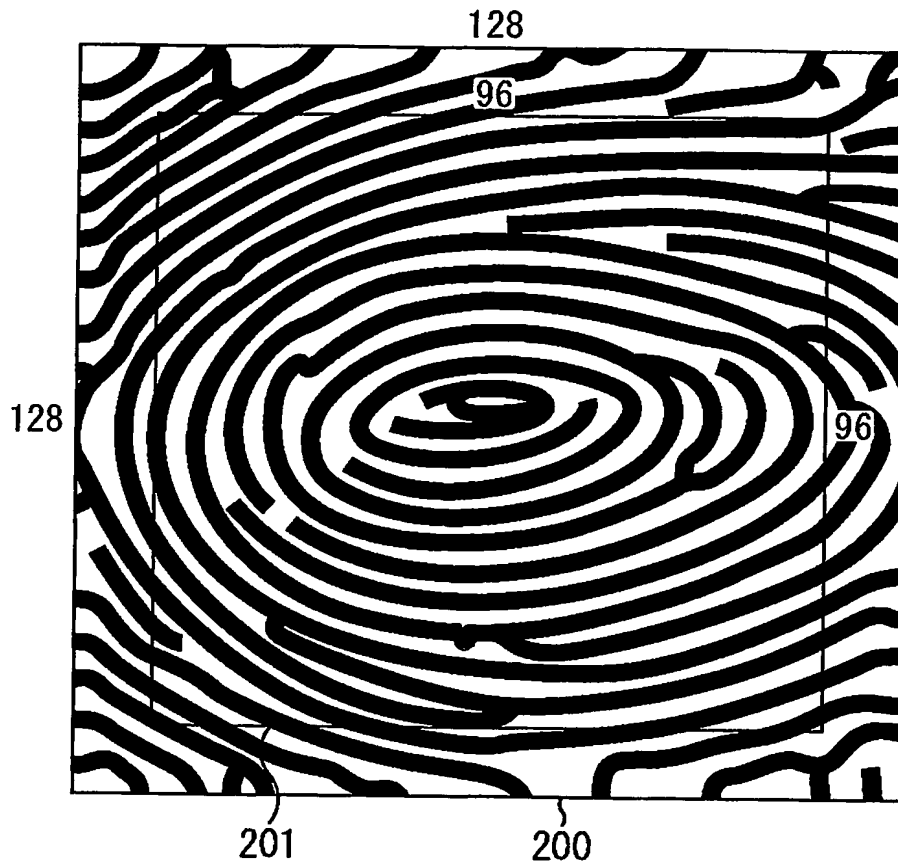
【図 9】

図9



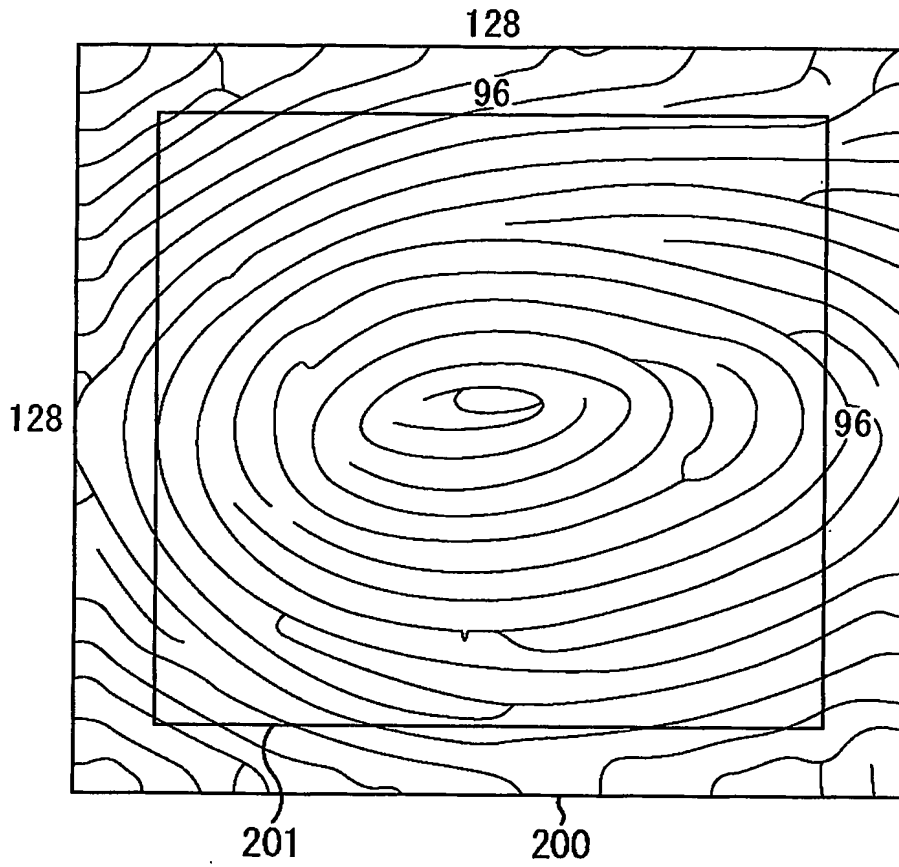
【図 10】

図10



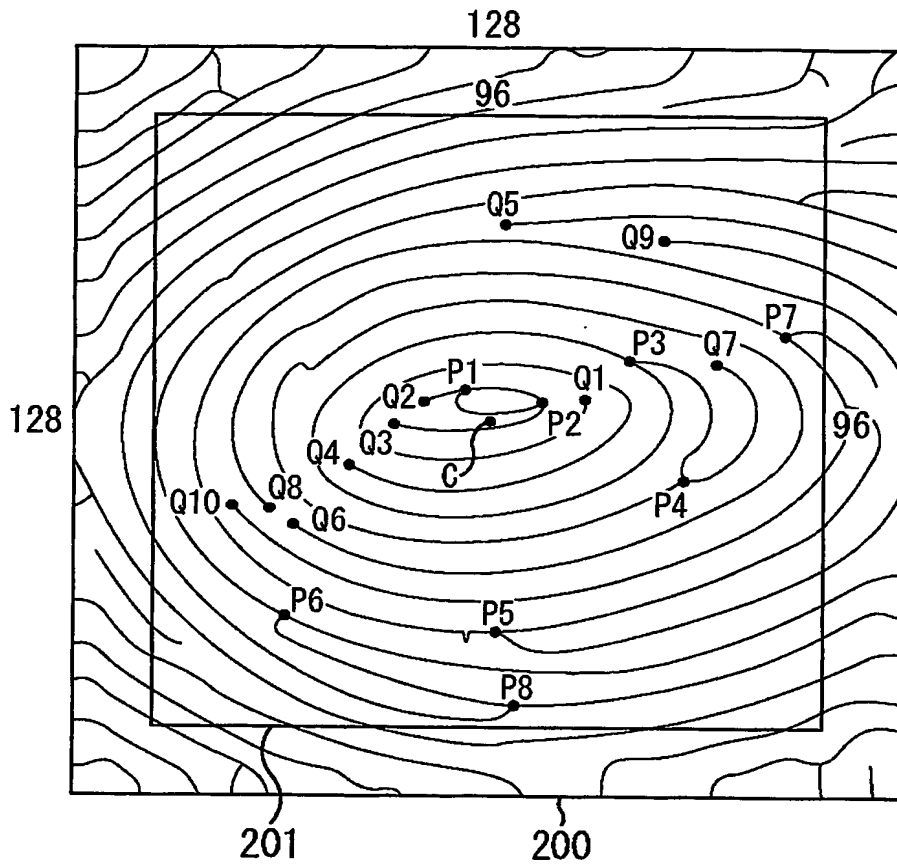
【図11】

図11



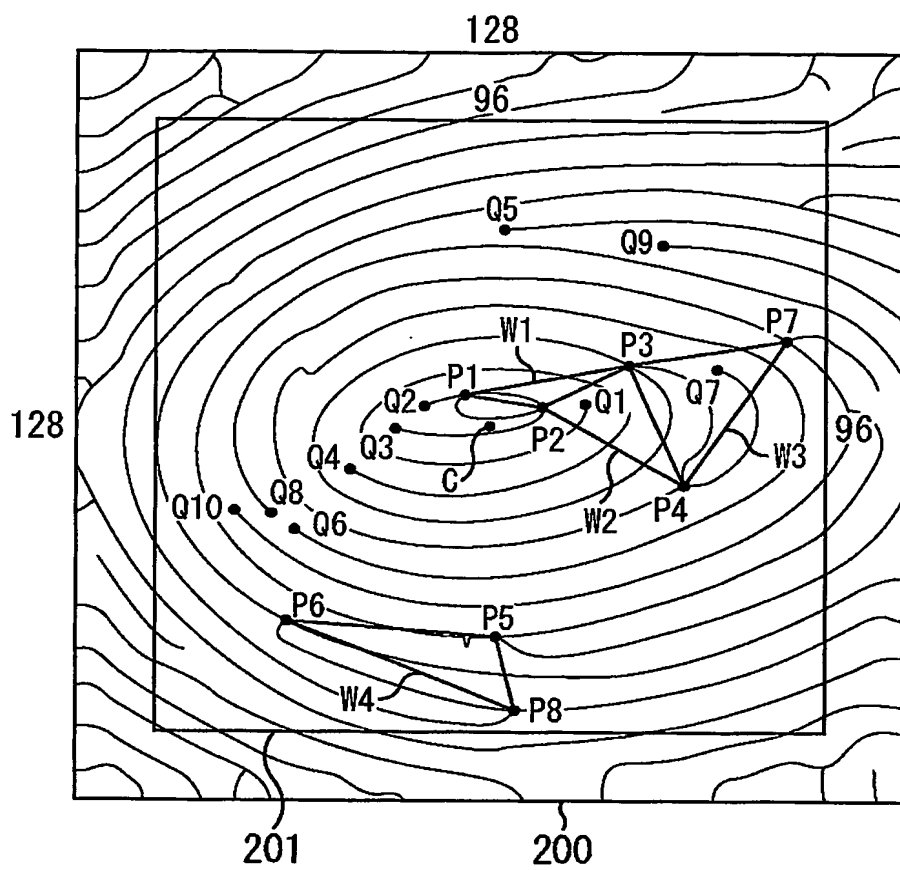
【図 12】

図12



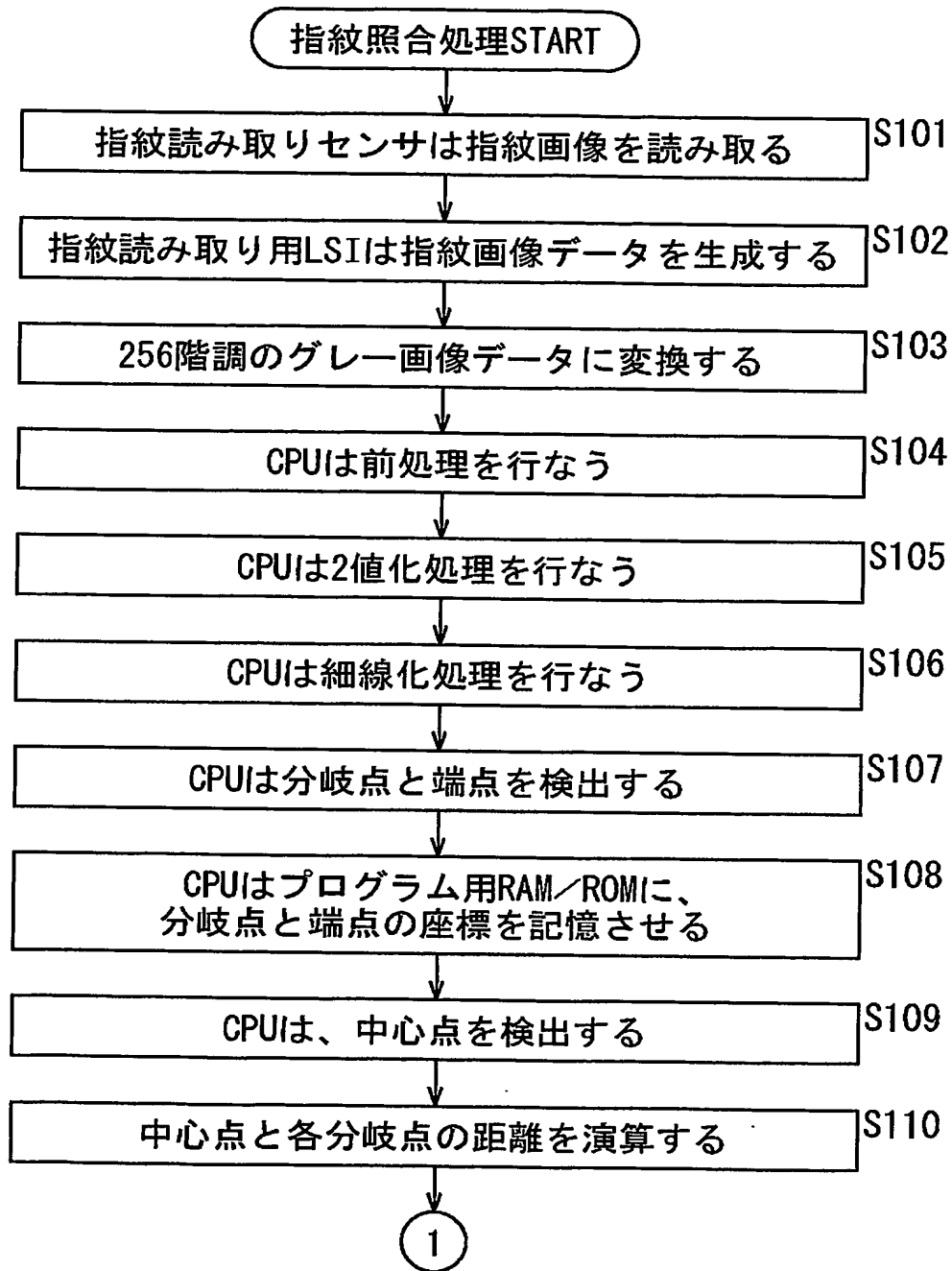
【図 13】

図13



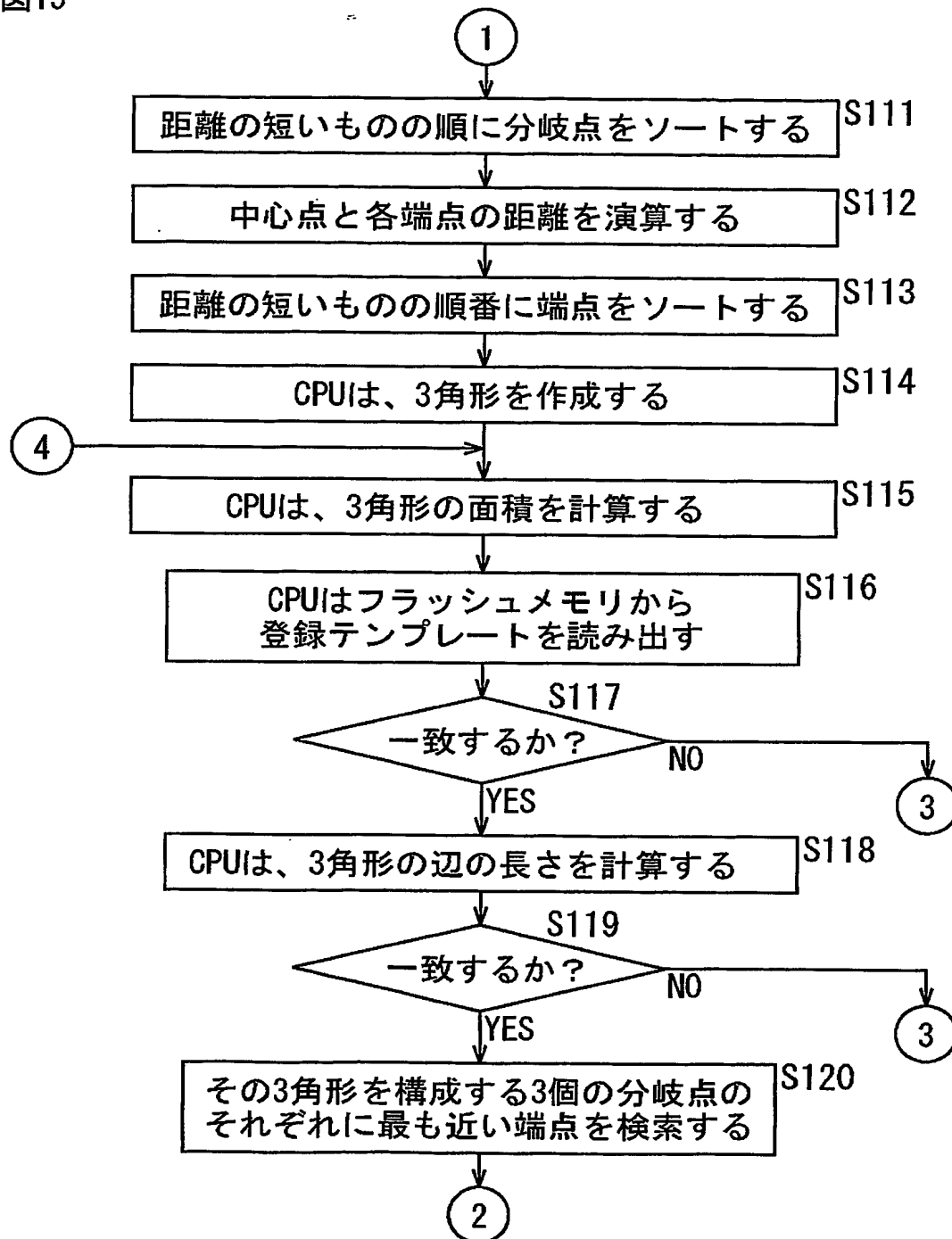
【図 14】

図14



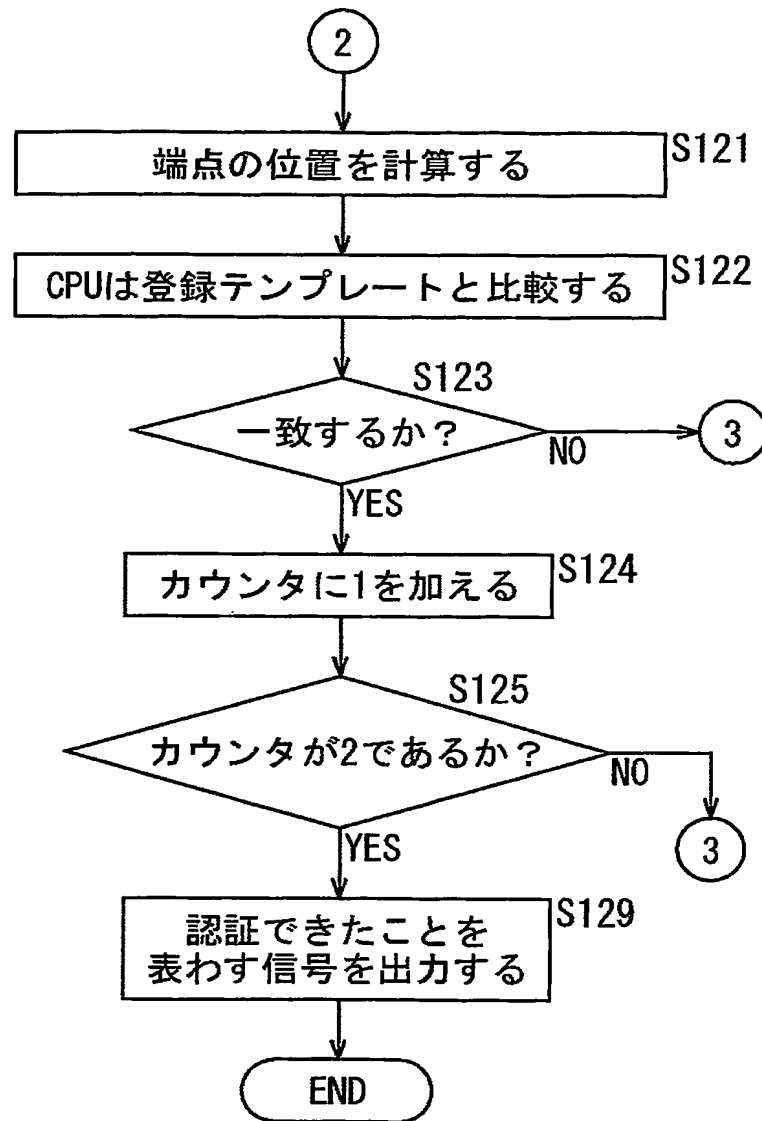
【図15】

図15



【図 16】

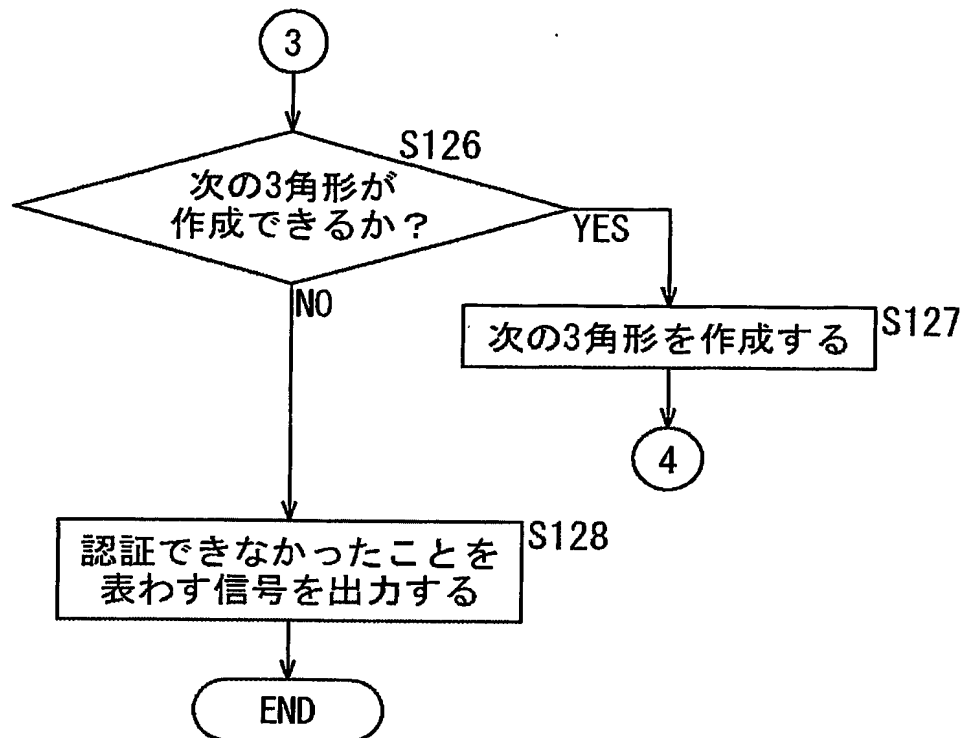
図16





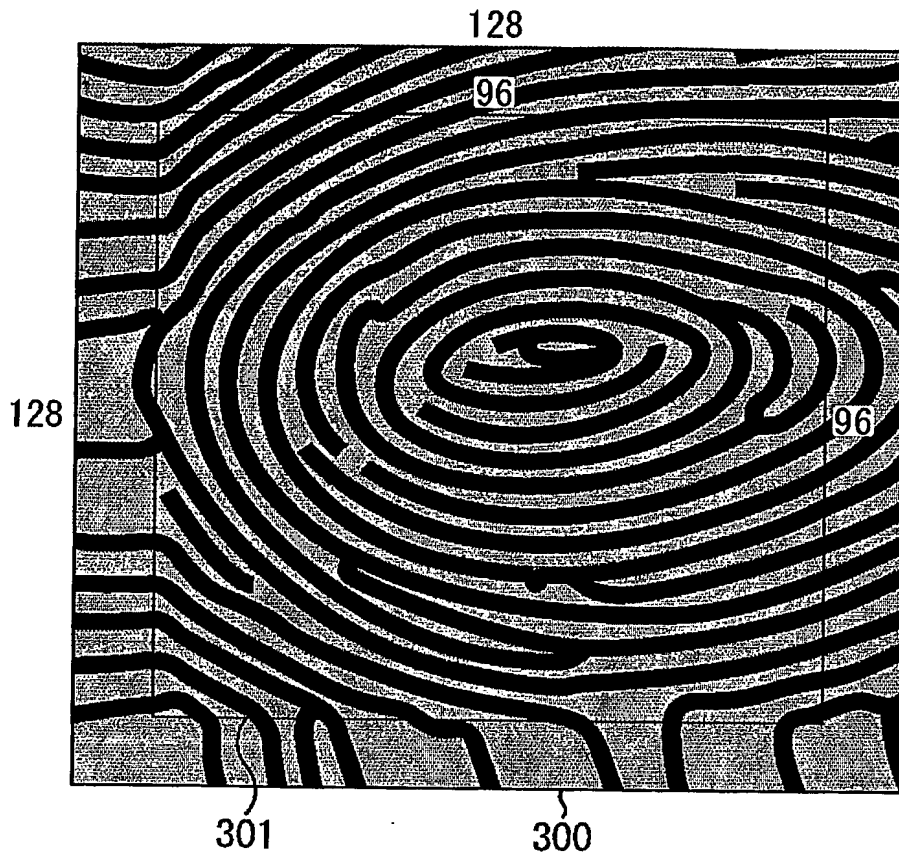
【図 17】

図17



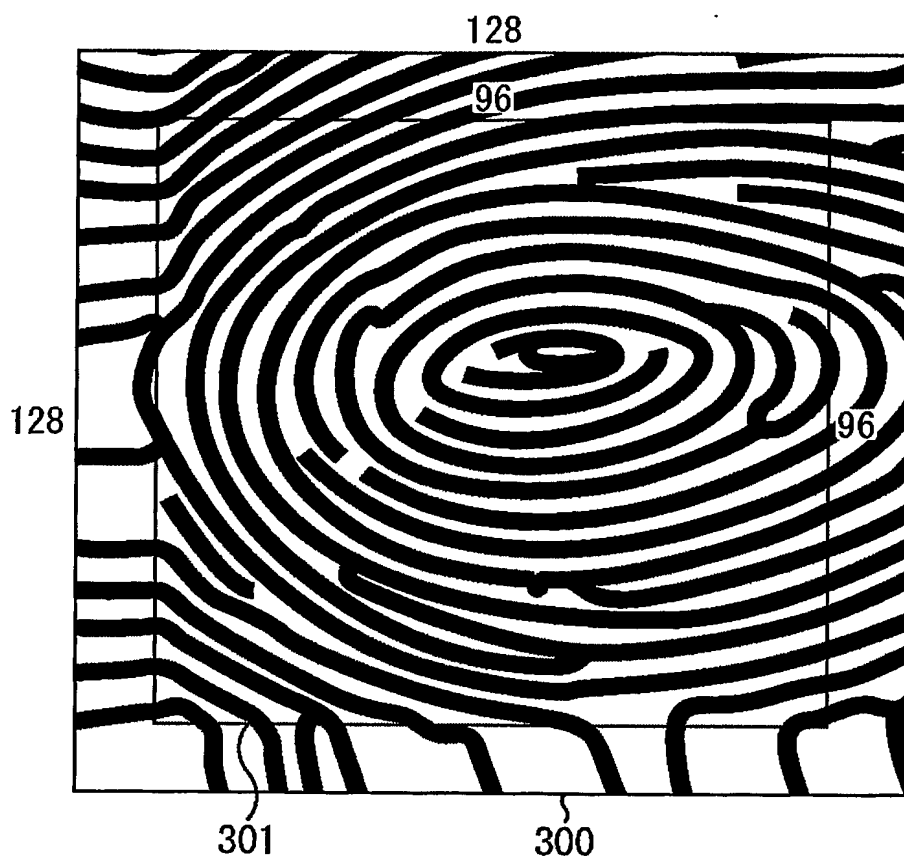
【図 18】

図18



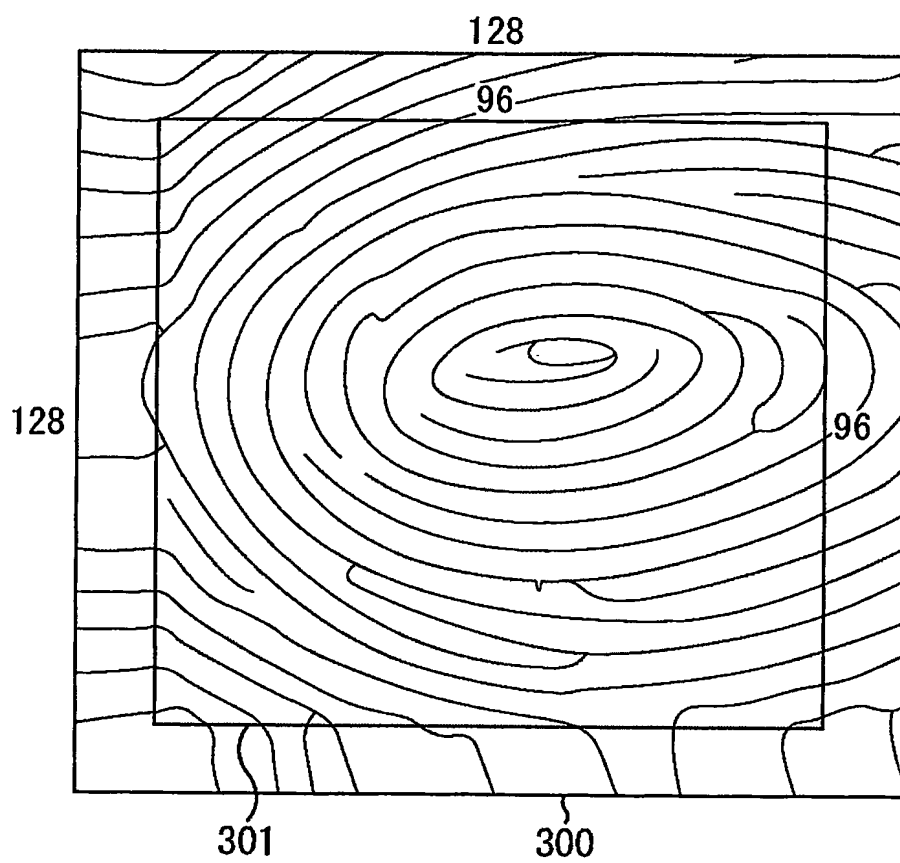
【図 19】

図19



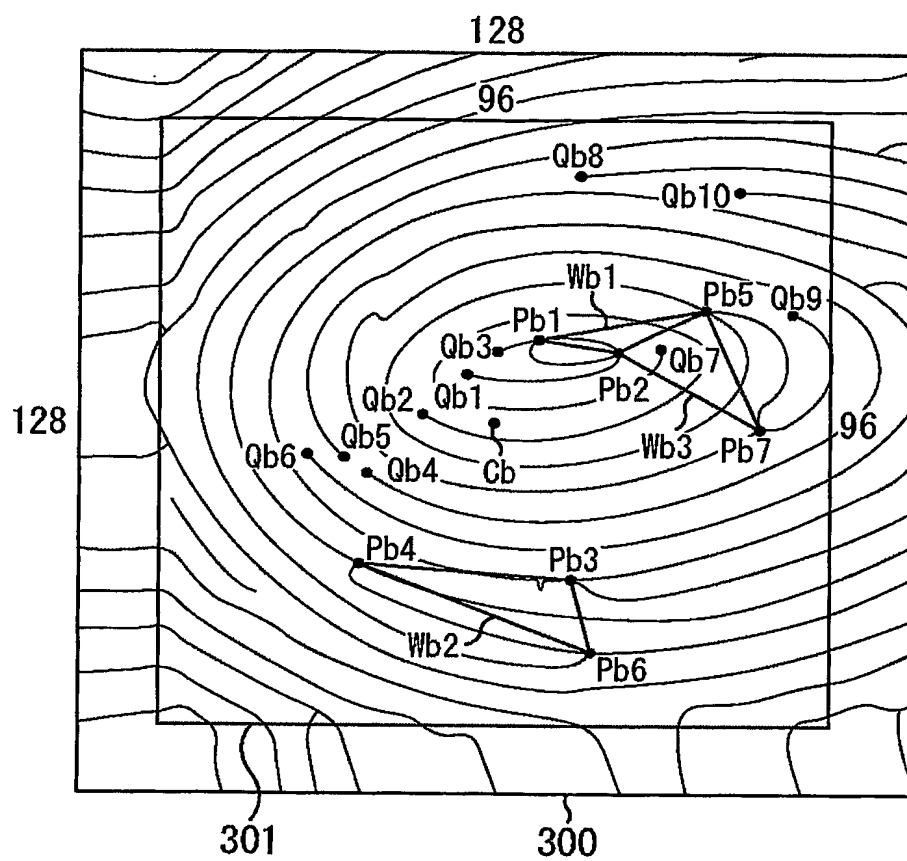
【図 20】

図20



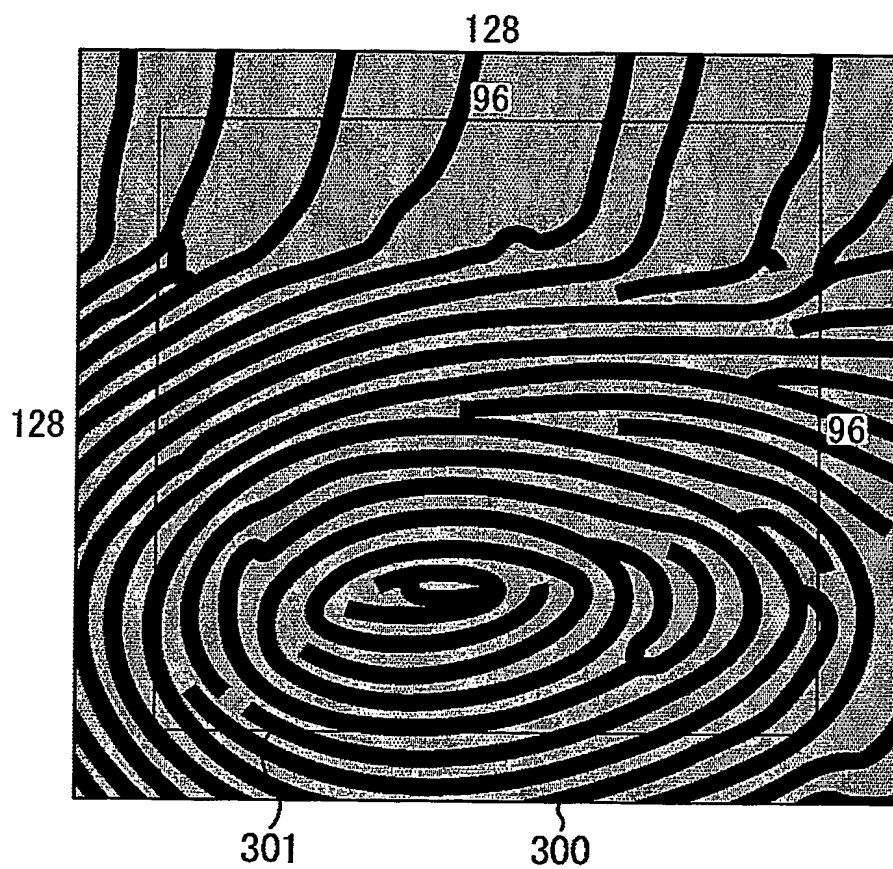
【図 21】

図21



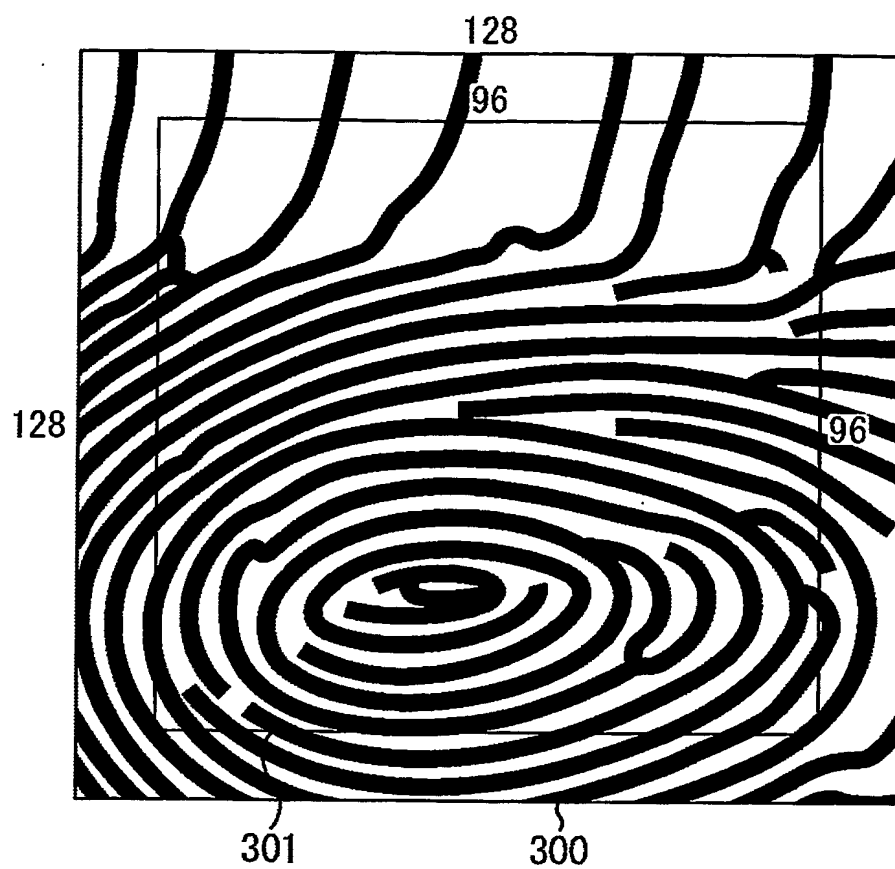
【図 22】

図22



【図 23】

図23



【図 24】

図24

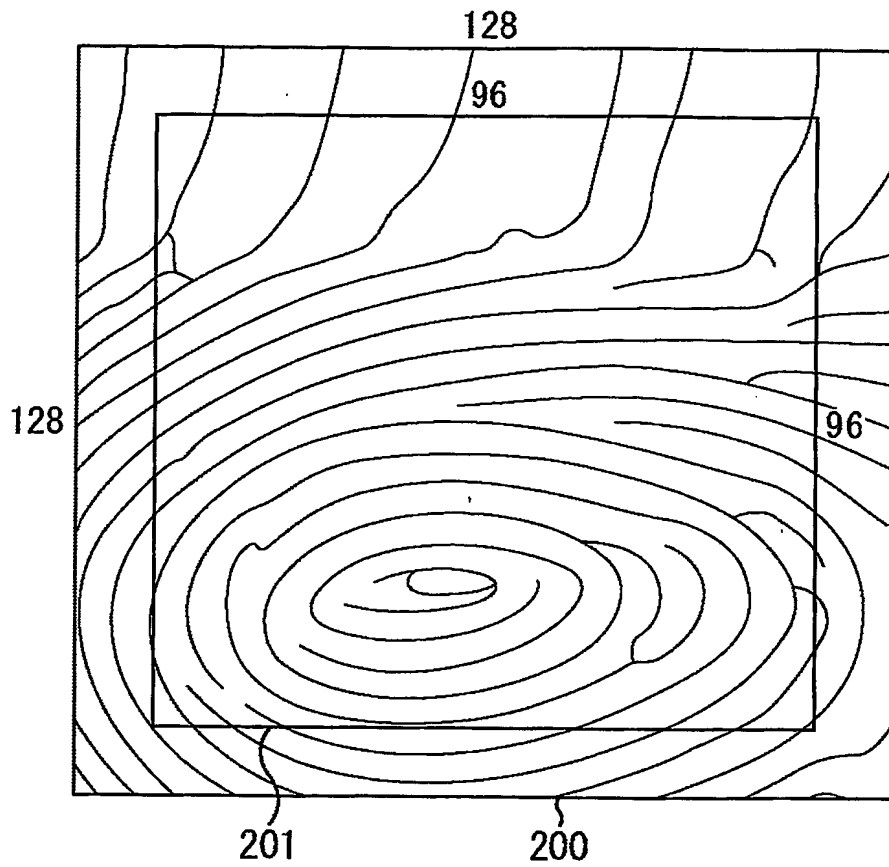
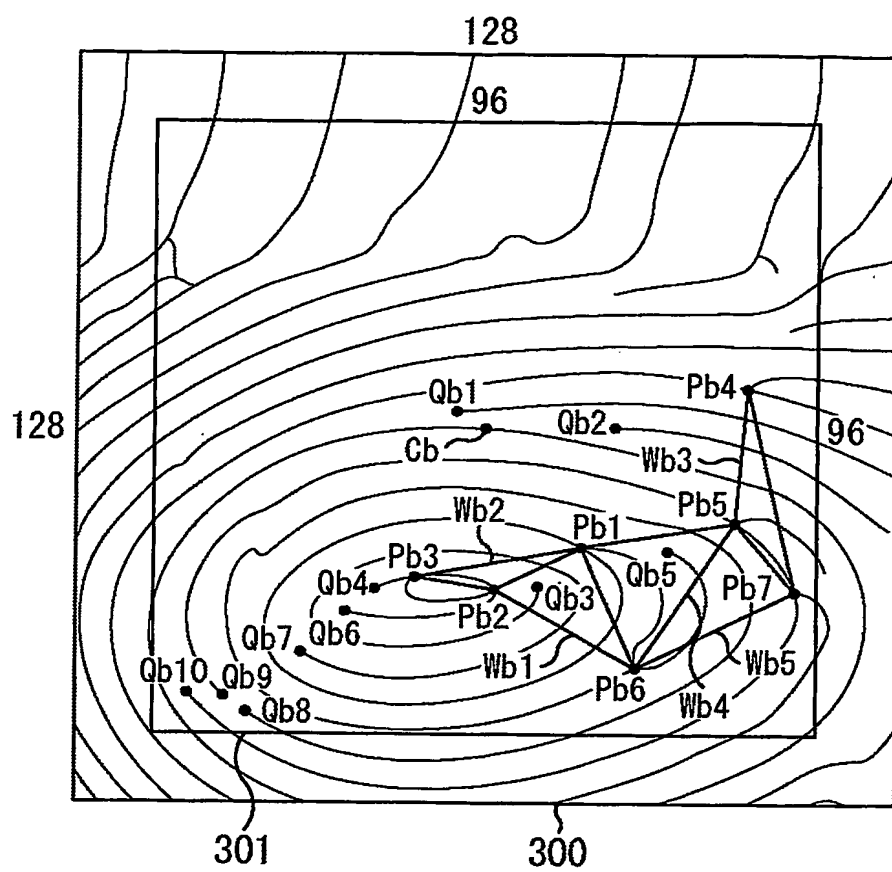




図25



【書類名】 要約書

【要約】

【課題】 指紋照合の処理速度を速く、かつ、データ量を少なくできるようにする。

【解決手段】 CPUは、指紋画像の中心点C、分岐点P1乃至P8、および端点Q1乃至Q10を検出し、中心点Cに最も近接する分岐点P1と、分岐点P1に近接する2つの分岐点P2、P3を結んで3角形W1を作成し、3角形の面積S1と3辺の長さをフラッシュメモリに記憶する。また、CPUは、分岐点P1と分岐点P1に近接する端点Q1、分岐点P2と分岐点P2に近接する端点Q3、および分岐点P3と分岐点P3に近接する端点Q3の位置をそれぞれ計算し、フラッシュメモリに登録テンプレートとして記憶する。照合時、CPUは、照合画像から同様に求めた3角形の面積と辺の長さ、端点との位置に基づいて、登録テンプレートと一致するか否かを判定する。本発明は、指紋照合装置に適用できる。

【選択図】 図13

特願 2003-000120

出 願 人 履 歷 情 報

識別番号

[000002185]

1. 変更年月日

1990年 8月30日

[変更理由]

新規登録

住 所

東京都品川区北品川6丁目7番35号

氏 名

ソニー株式会社